(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 1 145 244 B1

(12) **EUROPEAN PATENT SPECIFICATION**

(45) Date of publication and mention of the grant of the patent:
03.10.2007 Bulletin 2007/40

(21) Application number: **00950068.7**

(22) Date of filing: **27.07.2000**

(51) Int Cl.:
H04L 12/28 *(2006.01)* ***H04L 29/06*** *(2006.01)*
G06F 13/00 *(2006.01)* ***G06F 9/44*** *(2006.01)*

(86) International application number:
PCT/KR2000/000820

(87) International publication number:
WO 2001/008150 (01.02.2001 Gazette 2001/05)

(54) **DEVICE CUSTOMIZED HOME NETWORK TOP-LEVEL INFORMATION ARCHITECTURE**

INFORMATIONSARCHITEKTUR HÖCHSTEN NIVEAUS FÜR EIN AN GERÄTE ANGEPASSTES HAUSNETZWERK

ARCHITECTURE D'INFORMATION DE NIVEAU SUPERIEUR POUR RESEAU DOMOTIQUE PERSONNALISEE EN FONCTION DE DISPOSITIFS

(84) Designated Contracting States:
DE FR GB

(30) Priority: **27.07.1999 US 146101 P**
17.08.1999 US 149515 P
12.06.2000 US 592598

(43) Date of publication of application:
17.10.2001 Bulletin 2001/42

(73) Proprietor: **SAMSUNG ELECTRONICS CO., LTD.**
Suwon City, Kyungki-do 442-373 (KR)

(72) Inventors:
- **WANG, Dongyan**
  Santa Clara, CA 95054 (US)
- **HUMPLEMAN, Richard**
  Fremont, CA 94539 (US)

(74) Representative: **Grünecker, Kinkeldey,**
Stockmair & Schwanhäusser
Anwaltssozietät
Maximilianstrasse 58
80538 München (DE)

(56) References cited:
EP-A- 0 838 768
EP-A1- 0 833 479
EP-A2- 0 140 077
EP-A2- 0 457 673
WO-A-98/59282
WO-A-99/57839
JP-A- 9 261 355
JP-A- 9 282 263
JP-A- 11 088 406
JP-A- 11 194 987
JP-A- 11 317 756
JP-A- 11 355 294
JP-A- 11 355 357
US-A- 4 860 006
US-A- 5 392 033

EP 1 145 244 B1

Printed by Jouve, 75001 PARIS (FR)

Description

[0001] The present invention relates to the field of networks, and more particularly, to home networks having multimedia devices connected thereto.

[0002] A network generally includes a communication link and various devices with communication capability connected to the communication link. The devices include computers, peripheral devices, routers, storage devices, and appliances with processors and communication interfaces. An example of a network is a home network for a household in which various devices are interconnected. A usual household can contain several devices including personal computers and home devices that are typically found in the home. As such the term "device" typically includes logical devices or other units having functionality and an ability to exchange data, and can include not only all home devices but also general purpose computers. Home devices include such electronic devices as security systems, theater equipment, TVS, VCRs, stereo equipment, and direct broadcast satellite services or (DBSS), also known as digital satellite services (DSS), sprinkler systems, lighting systems, micro waves, dish washer, ovens/stoves, washers/dryers, and a processing system in an automobile.

[0003] In general, home devices are used to perform tasks that enhance a homeowners life style and standard of living. For example, a dishwasher performs the task of washing dirty dishes and relieves the homeowner of having to wash the dishes by hand. A VCR can record a TV program to allow a homeowner to watch a particular program at a later time. Security systems protect the homeowners valuables and can reduce the homeowners fear of unwanted entry.

[0004] Home devices, such as home theater equipment, are often controlled using a single common control unit, namely a remote control device. This single common control unit allows a homeowner to control and command several different home devices using a single interface. Thus, many manufacturers have developed control units for controlling and commanding their home devices from a single interface.

[0005] One drawback associated with using the remote control unit to command and control home devices is that it provides static and command logic for controlling and commanding each home device. Therefore, a particular remote control unit can only control and command those home devices for which it includes the necessary control and command logic. For example, if a remote control unit comprises logic for controlling a television (TV), a video cassette recorder (VCR), and a digital video device (DVD), but not a compact disk (CD) unit, the remote control unit can not be used to command and control the CD unit. In addition, as new home devices are developed, the remote control unit will not be able to control and command the new home devices that require control and command logic that was not known at the time the remote control unit was developed.

[0006] Further, typically a remote control unit can only be used to command and control those home devices that are within the signal range of the remote control unit. Therefore, a user cannot use the remote control unit from a single location in the house to control and command home devices that are interconnected, but located in separate areas of the home. For example, a VCR that is located upstairs in a bedroom may be connected to a TV that is downstairs in the family room. If a user wishes to play a tape contained in the VCR located upstairs in the bedroom, on the TV located downstairs in the family room, the user cannot control and command both the TV and the VCR from a single location.

[0007] Another drawback associated with using remote control units is that known remote control units cannot control a plurality of diverse devices, and more particularly, cannot control a plurality of devices having different capabilities to communicate with each other in order to accomplish tasks or provide a service. Further, conventional network systems do not provide a mechanism for software applications in different network devices to automatically communicate with one another in order to accomplish tasks without direct user command.

[0008] To alleviate the above problems, some network models provide a central/singular user interface (UI) in one device including static device information for networked devices for user control of network devices. However, in such networks a change to device information (e.g., ICON) in a device requires a change to, and rebuilding of, the top level page. Further, if the device displaying the central user interface becomes unavailable, user control of the network is curtailed. Another problem with the central/singular page is that every UI device must display the same page, and a scope is not provided for each manufacturer to generate its own UI look and feel nor alter the technology used in the UI device. The content of an icon/information representing a device cannot be changed, and a UI device cannot display a more prominent look to a device icon such as the icon for the UI device itself. Nor can a UI builder tool obtain e-business icons from an external Web Portal. Such a model cannot be standardized for industry use because a central/single UI device controls the UI.

[0009] There is, therefore, a need for a method and a system which provides dynamic control and command devices in a home network. There is also a need for such a method and system to provide the ability for one or more devices connected to the network to independently generate different user interface representations of the devices connected to the network for user command and control.

[0010] WO 98 59282 A relates to a home network to which a number of home devices are currently connected. The home devices includes clients, i.e., devices providing control interface service to a human operator, including a graphical display hardware, and server devices, i.e., modules supplying a service which may be any service other than a control

interface provided by a client. Each home device connected to the home network has one or more associated HTML files. The HTML files for a respective home device define the control and command functions for that particular home device. Further, the HTML files contain data that enables a browser to present the information contained in the HTML files graphically. By conforming to the HTML and HTTP Internet standards, each home device sends its custom GUI to a browser based DTV acting as a client. The browser based DTV receives and interprets the HTML files associated with the home devices acting as servers and graphically displays the respective control and command information on its viewable display. Once the information contained in a device's HTML file is graphically displayed on the DTV, the user can control the home device from the DTV by selecting icons that have associated hyperlinks to start the control programs displayed on the DTV's screen. More particularly, control may be implemented by transfer of a graphical control object (GCO), which preferably resides in the server, from the server for rendering on the client, to make the GUI. The look and feel of the GUI originates with the attached server and not with the client.

[0011] Post-published WO 99 57839 A discloses methods and systems for performing a service on a home network having a plurality of home devices connected thereto. A client device is connected to the home network for displaying a user interface. A software agent on the client device is executed for obtaining selection information for the network devices and displaying the selection information on a user interface displayed on the client device. The first home device connected to the network is selected from the user interface being displayed on the client device. First capabilities data for the first home device and second capabilities data for a second home device connected to the network are read. The first and second capabilities data includes information in a structured format for identifying the capabilities of the first and second home device, respectively. The first and second capabilities data of the first and second home devices, respectively, are compared. The second home device is selected from the user interface displayed on the client device. Control and command data is sent from the client device to the first and second home devices to cause the first and second home devices to communicate with each other to perform the service.

[0012] It is the object of the present invention to provide a technique for providing a user interface for controlling devices in a network containing more than one client.

[0013] This object is solved by the invention as defined in the independent claims. Embodiments are given in the dependent claims.

[0014] In one embodiment, the present invention provides a method and system for generating a user interface in a plurality of multiple devices connected to the network system for controlling devices that are currently connected to a network. In one version the network system includes a physical layer, wherein the physical layer provides a communication medium than can be used by devices to communicate with each other, and multiple devices connected to the physical layer, one or more of said multiple devices storing information including device information. A plurality of said multiple devices each including an agent adapted for: a) obtaining information from devices currently connected to the network, said information including device information; b) generating a user interface description in each of said one or more devices based at least on the obtained information, the user interface description in each device including at least one reference associated with the device information of each of said devices currently connected to the network; and c) displaying one or more user interfaces each based on one of said one or more user interface descriptions, on one or more devices connected to the network capable of displaying a user interface, for user control of said devices that are currently connected to the network.

[0015] In one example, network devices top-level UI description is generated independently by any network device and certainly by devices capable of displaying UI (UI device). Generating a user interface in each device rather than generating a centrally UI, allows a device to show its own device icon/text preferentially in the GUI. In addition each GUI is manufacturer customizable, user configurable and also more reliable because it does not depend on another device e.g. a single central server.

[0016] These and other features, aspects and advantages of the present invention will become better understood with regard to the following description, appended claims and accompanying drawings where:

FIG. 1 shows an example block diagram of the architecture of an embodiment of a network according to the present invention;

FIG. 2 shows an example block diagram of the architecture of another embodiment of a network according to the present invention;

FIG. 3 illustrates an example of a layered interface model that can be used for communicating between home devices in accordance with the present invention;

FIG. 4A shows an example architecture diagram of a DVCR server device replaying video to a DTV client device capable of displaying a user interface, in a network according to the present invention;

FIG. 4B shows another example architecture diagram of a server device communicating with a client device capable of displaying a user interface, in a network according to the present invention;

FIGS. 5-8 illustrate example top-level GUIs representing the functions of networked devices to a user;

FIG. 7 shows an example block diagram architecture of a home network constructed in accordance with another

embodiment of the present invention;

FIG. 8 shows an example process according to the present invention for communication between a 1394 network and a non-1394 network for IP address configuration;

FIGS. 9A-C show example functional block diagrams of connections to data and control bits of an embodiment of a discovery system architecture in a network according to another aspect of the present invention;

FIG. 10 shows an example flow diagram for discovery and configuration agents in the home network in connection with the functional block diagrams in FIGS. 9A-C;

FIG. 11 shows an example flow diagram for user interface agent in the home network in connection with the functional block diagrams in FIGS. 9A-C; and

Appendices 1-4, illustrative examples for: 1) Top-Level Page description 250 (Appendix 1); 2) Background.htm (Appendix 2); 3) Icon.htm (Appendix 4); and 4) Name.htm (Appendix4).

[0017] To facilitate understanding, identical reference numerals have been used, where possible, to designate identical elements that are common throughout the figures.

[0018] Referring to FIG. 1, in an embodiment of the present invention, a network 10 comprises multiple devices 11 including at least one client device 12 and at least one server device 14 interconnected via a communication link 16. The communication link 16 can include a 1394 serial bus providing a physical layer (medium) for sending and receiving data between the various connected home devices. The 1394 serial bus supports both time-multiplexed audio/video (A/V) streams and standard IP (Internet Protocol) communications (e.g., IETF RFC 2734). In certain embodiments, a home network uses an IP network layer as the communication layer for the home network. However, other communication protocols could be used to provide communication for the home network. For example, the invention may be implemented using Function Control Protocol (FCP) as defined by IEC 61883, or any other appropriate protocol. Thus, a network may generally include two or more devices interconnected by a physical layer exchange or transfer of data in accordance with a predefined communication protocol.

[0019] Each client device 12 may communicate with one or more server devices 14 in the network 10. Further, each server device 14 may communicate with one or more other server devices 14, and one or more client devices 12, in the network 10. Each client device 12 can include a user communication interface including input devices such as a mouse and keyboard for receiving user input, and a display for providing a control user interface for a user to interact with the networked devices. The user interface can include a graphical user interface (GUI) 18 for providing information to the user. Each server device 14 includes hardware as a resource in the network for providing services to the user, and can further include a server or service control program 20 for controlling the server hardware.

[0020] Each server device 14 provides a service for the user, except control user interface, and each client device 12 provides a service including control user interface for user interaction with the network 10. As such, only client devices 12 interact directly with users, and server devices 14 interact only with client devices 12 and other server devices 14. Example services can include MPEG sourcing/sinking and display services.

[0021] In an exemplary embodiment of the present invention, a browser based network (e.g., a home network) uses Internet technology to control and command devices including client devices and server devices that are connected to a network. Each device includes device information such as interface data (e.g. HTML, XML, JAVA, JAVASCRIPT, GIF, JPEG, graphics files, or any other format useful for the intended purpose) that provides an interface for commanding and controlling of the device over the network. In certain embodiments, each device includes device information such as one or more Hypertext markup Language (HTML) pages that provide for the commanding and controlling of that device. Using the browser technology, the network employs Internet standards to render the HTML pages in order to provide users with a plurality of graphical user interface (GUIs) for commanding and controlling each device. In one example, the network is configured as an Intranet.

[0022] In one embodiment, a client device comprises a device providing control interface service to a human operator, including a graphical display hardware for down communication and a mouse or other point-and-click device for up (or return) communication. A server device comprises a module supplying a service, which can be any service other than a control interface provided by a client device. As such, the server/client device relationship is a control relationship, wherein the server device provides a service but a client device can use the data, as a DTV displays video data, but need not manipulate or alter the data. It is thus consistent with this definition to observe that, frequently, a server may be a source of information and a client (a browser, for example) may be a consumer of information.

[0023] Examples of specific functions which can be implemented by server devices include: return of information (data); performance of a function (e.g., mechanical function) and return of status; return of a data stream and status; reception of a data stream and return of status; or saving of a state for subsequent action. Examples of server devices include MPEG source, sink and display servers. While a server device typically includes a custom, built-in, control program to implement control of its own hardware, a client functions to interface with the server device. However, server device as used herein does not imply that a web server and a protocol stack must be used.

[0024] FIG. 2 shows a block diagram of an embodiment of a network 100 according to an aspect of the present

invention. A 1394 serial bus 114, described above, electronically connects multiple devices 11 including server devices 14 (e.g., DVD 108, DVCR 110), client devices 12 (e.g., DTV 102, 103), Bridge 116, DVCR120, PC 105, cable/modem access 107, and DBS access 109, on the network 100. FIG. 3 illustrates an example of a layered interface model that can be used for communicating between the devices 11 in accordance with the present invention. In this example, a device (server) 150 communicates with a client device 166 using one or more of the network communication layers 152-164. In one example, an application in the device 150 communicates with an application in the device 166 via the network layer 160. The details of lower layers 162 and 164 are not seen by the applications, whereby use of e.g. either 1394 or Ethernet does not make a difference to said applications in the devices 150, 166. Further not all the upper layers of the 7-layer model are used all the time (e.g., in the Web model (TCP/IP model) session layer 156 and presentation layer 154 are not used). As such, in one version, by employing the Internet Protocol standard for the network layer 160, the devices can communicate with each other without having to know specific details about the other communication layers (i.e. application 152, presentation 154, session 156, transport 158, data link 162 and physical 164). Thus, by employing the Internet Protocol standard for the network layer 160, the network can use a combination of different communication layers in communicating between different devices.

[0025] A single physical package can include several devices which are logically networked via a network layer for example as shown in FIG. 3 not necessarily via a physical network (e.g., such devices can include a VCR and a TV in a single housing). Where a logical device accesses a GUI to enable a user to control a device, the device and the logical device can be included in the same physical package. In such an embodiment, the physical device fetches a GUI from itself. However, in other embodiments the network interconnects separate physical devices, wherein for example, a first device fetches a GUI from a second device, to permit user interaction with the GUI to control the second device.

[0026] In a presently preferred embodiment, a 1394 serial bus is used as the physical layer 164 for the data communications on the network 100. Because of its enhanced bandwidth capabilities (e.g., enhanced and guaranteed bandwidth and isochronous stream capability), the 1394 serial bus can provide a single medium for all data communications on the network 100 (i.e. audio/video streams and command/control).

[0027] Further, the 1394 serial bus provides automatic configuration reset such that when a device is plugged in/removed all the 1394 interfaces reset, the 1394 bus reconfigures and every device knows the presence of every other device (including a newly added one or without the one just removed). Also, the 1394 interface supports a data space for configuration information that is addressable from any device allowing other devices to write/read information and make modifications e.g. to permit the operation of the network layer protocol. However, it is possible to achieve these results with different software and standards. As such, the network 100 is not restricted to using a 1394 serial bus, and, in alternative embodiments of the present invention, other bus types, such a Ethernet, ATM, wireless, etc., can be used as the physical layer if they meet the particular throughput requirements of an individual network (e.g., a home network). Further, a modified version of e.g. wireless-Ethernet can include the essential features of 1394.

[0028] As depicted in FIG. 2, the network 100 includes several devices connected to the 1394 serial bus 114. In this example, the devices include a DBSS 104 for receiving transmission signal from a satellite 122 for subsequent display. Associated with the DBSS is a network interface unit ("NIU") which, among other things, provides an interface between the DBSS satellite transmission and the 1394 serial bus 114.

[0029] A digital video device (DVD) 108 is also connected to the exemplary network 100. The DVD 108 can be used to display digitally encoded videos on a television. Also connected to the exemplary network 100 is a digital video cassette recorder (DVCR) 110, i.e., a digital TV 102. In this example, the DTV 102 provides a human interface for the network 100 by employing browser technology to allow users to control and command for devices over the home network 100. A second DTV 103 provides another human interface for the network 100 by employing browser technology to allow users to control and command for devices over the home network 100. The DTVs 102 and 103 can provide human interfaces for the network 100 as each DTV comprises a screen for displaying HTML pages. However other devices having display capability can be used to provide human interfaces. Thus, in certain embodiments of the invention, a device such as the personal computer 105 (PC) is used to provide a human interface for a respective home network, as a PC 105 typically embodies a screen display unit.

[0030] The 1394 serial bus 114 is depicted as using the HTTP/IP interface protocol, and preferably HTTP/TCP/IP, wherein IP provides packet format (a one-way write only model), TCP provides an error free version of IP (e.g., ensures packets arrive and in correct order), and HTTP provides 2-way connection (packet to server will expect a response -a 'read' model). Certain devices can require other protocol interface types (e.g., UPD/IP, FTP/IP, TELNET/IP, SNMP/IP, DNS/IP, SMTP/IP). In certain embodiments of the invention, a proxy 116 can be used to interface two networks using dissimilar interface protocols on their respective mediums which, when connected, comprise the network 100. The proxy 116 (e.g. Web proxy) can include Home Automation type protocols such as the HTML/HTTP/TCP/IP proxy for X10, Lonworks, CEBus (on their respective physical technologies), or non-IP protocols on 1394 (e.g., AVC/FCP/1394).

[0031] In certain embodiments, the two network mediums are of the same type. For example, as depicted in FIG. 2, the 1394 serial bus 114 using the HTTP/IP interface protocol is connected by a proxy 116 to the Home Automation neutral 118 (e.g., X10). By using the proxy 118 as HTML/HTTP/CTP/IP/1394 proxy for VCR-Commands/AVC/FCP/

1394, to interface between HTML/HTTP/TCP/IP and X10 protocols, DVCR 120 is also accessible on the network 100. In certain other embodiments, a network can comprise two network mediums of dissimilar types, e.g., a 1394 Serial bus and Ethernet. Therefore, in certain embodiments of the invention, a proxy is used to interface two dissimilar medium types to from a single network. A discovery process, described further below, can be used for the discovery of devices that are powered on and connected to the network 100. Also, the same 1394 bus can be used without need for a bridge box.

[0032] As depicted in FIG. 2, devices 11 including DTV 102, DTV 103, PC 105, DVCR 110, DVD 108, DSS-NIU 104 and DVCR 120 represent devices that are currently connected to the network 100 comprising a 1394 network. A client-server relationship exists among the attached devices, with the DTV 102 , DTV 103 and PC 105 typically behaving as clients and devices DVCR 110, DVD 108, DSS-NIU 104 and DVCR 120 behaving as servers.

[0033] A typical 1394 network comprises interconnected devices such as a collection of appliances including server devices offering one or more services to be controlled (e.g., DVCR 100 as an MPEG video recording and replay service), and client device offering a user interface (UI) service (e.g., DTV 102) for controlling the server devices. Some appliances (e.g., DTV 103) can have both services (e.g., MPEG decode and display capability) to be controlled, and a UI controller capability. According to an aspect of the present invention, methods and systems including protocols, document description, image compression and scripting language standards from technologies utilized in the World Wide Web standard (Web model) are used to implement t a 1394WEB user-to-device control model in the network 100. The Web model is a client/server model. The controlled server device (service) comprises a Web server and the controller client device (i.e., a device capable of displaying a UI) comprises a Web client including a GUI presentation engine, described further below, such as a Web browser (e.g., Explorer™ , Netscape™, etc.).

[0034] FIG. 4A shows a server device such as the DVCR 110 replaying MPEG video to a client device such as the DTV 102 in a network 100 according to the present invention, wherein the DTV 102 can display a user interface. The DVCR 110 includes Web server hardware and software and the DTV 102 includes Web browser software. A user can utilize the DTV 102 to request that the DTV 102 display a user interface based on the device information 202 contained in the DVCR 110 or based on the device information 204 contained in the DTV 102. For example, the user can utilize a browser 200 in the DTV 102 to display an HTML control page GUI 202 contained in the DVCR 110 or an HTML control page GUI 204 contained in the DTV 102. Each page 202, 204 includes graphical user interface description information in HTML, wherein the browser 200 reads that information to generate a graphical user interface. Each page 202, 204 represents the Control Interface of the Applications 206, 212, respectively. Each page 202, 204 can include a hierarchy of pages to represent a corresponding application control interface.

[0035] Each GUI 202 and/or 204 includes active control icons and/or buttons for the user to select and control devices currently connected to the network 100. If, for example, the user selects a PLAY button in the GUI 202 of the DVCR 110 displayed by the browser 200 on the DTV 102, a hyperlink message is returned to the DVCR 110 Web server and directed to an application software 206 (e.g., MPEG Record/Replay Service Application Software) in the DVCR 110 for operating a DVCR hardware 208. In one example, an MPEG video stream source 208 in the DVCR 110 transmits an MPEG video stream to an MPEG vide decode and display system 210 in the DTV 102 for display under the control of application control software 212 in the DTV 102. The application software 206 in the DVCR 110 also sends information back to the application software 212 in the DTV 102, including e.g. an acknowledgment if the operation is successful, or an altered or different control GUI 202 to the DTV 102 indicating status to the user. There can be further communication between the application softwares 206 and 212 e.g. for setting up a 1394 isochronous video stream connection for video stream service.

[0036] FIG. 4B shows another example architecture diagram of a server device communicating with a client device capable of displaying a user interface, in a network 100. The server device such as DVCR 110 replays MPEG video to the client device such as the DTV 102 in the network 100, wherein the DTV 102 can display a user interface.

[0037] In an embodiment of the invention, the communication protocol between devices in the network 100 is based on the Hypertext Transfer Protocol (HTTP 1.1), an application-level protocol for distributed, collaborative, hypermedia information systems. HTTP is a generic, stateless, object-oriented protocol that can be use for many tasks. A feature of HTTP is the typing and negotiation of data representation, allowing devices to be built independently of the data being transferred over the network 100 to which the devices are connected.

[0038] The description document language for defining various GUIs 202, 204 can be e.g. HTML, version 4.0, the publishing language of the World Wide Web. HTML supports text, multimedia, and hyperlink features, scripting languages and style sheets. HTML 4.0 is an SGML application conforming to International Standard ISO 8879 - Standard Generalized Markup Language.

[0039] To display images, three still image graphics compression formats specified by the HTML specification are utilized in the 1394WEB network 100 for ICON, LOGO and other graphics. The still image graphics compression formats are: Graphics Interchange Format (GIF89a), Progressive Joint Photographic Experts Group (JPEG) and Portable Network Graphics (PNG). Table 1 shows the differences in capabilities between the three different still image graphics compression formats.

<Table 1: Still Image Compression Formats>

	PNG	Progressive JPEG	GIF89a
Color Depth	48 bit	24 bit	8 bit
Colors Supported		16.7 million	256
Formats Supported	Raster, Vector	Raster	Raster
Compression Scheme	LZ77 derivative	JPEG	LZW
Transparency	Per Pixel for Grayscale & RGB, Per Color for Indexed, 256 levels	No	Single Color, 2 levels (Binary)
Progressive Display	Yes	Yes	Yes
Scalable	No	No	No
Animation		No	Yes
Lossless Compression	100%		
Truecolor	48 bits		
Grayscale	16 bits		
Indexed-color	yes		
Gamma Correction (light intensity)	Yes		
Chromaticity Correction	Both		
Searchable Meta-Data	Yes		
Extensibility	Yes, chunk encoded		

[0040] Further, the Web scripting language, ECMA-Script-262, is utilized to provide a means for visually enhancing the GUI Web pages 202 as part of a Web-based client-server architecture. The scripting language is a programming language for manipulating, customizing, and automating the facilities/services of the devices. The user interface 200 provides basic user interaction functions, and the scripting language is utilized to expose that functionality to program control. The existing system provides the host environment of objects and facilities completing the capabilities of the scripting language. The web browser 200 provides the ECMA-Script host environment for client-side computation including, for example, objects that represent windows, menus, pop-ups, dialog boxes, text areas, anchors, frames, history, cookies, and input/output.

[0041] The web browser 200 provides the host environment for the EXMA-Script-262, and the host environment supports attaching scripting code to events such as change of focus, page and image loading, unloading, error and abort, selection, form submission, and mouse actions. Scripting code is included within the HTML pages 202 and 204 and the displayed page is the browser 200 includes a combination of user interface elements, and fixed and computed text and images. The scripting code responds to user interaction without need for a main program.

[0042] In one example, the specification for a 1394WEB client browser 200 includes HTTP1.1 specification, wherein section '8.1.2.1 Negotiation' of the HTTP1.1 specification regarding connection persistence is modified such that an HTTP1.1 client device such as e.g. the DTV102 expects a connection to server device such as e.g. the DVCR 110 via the 1394 to remain open, because the persistent connection in 1394WEB user control allows full status reporting from the server device (DVCR 110) while the GUI 202 and/or 204 remains visible in the browser 200 of the client device (DTV 102). The HTTP connection remains open (HTTP spec RFC 2068) wherein a client that supports persistent connections may "pipeline" its requests (i.e., send multiple requests without waiting for each response). A server must send its responses to those requests in the same order that the requests were received. This allows the web browser 200 to pipeline requests to the DVCR 110 which the DVCR 110 can then satisfy later with e.g. status responses such as Now Playing, Now Recording, Rewind Finished, Tape Broken, Etc. Other example implementations include e.g. the control page from the DVCR 110 can contain a request to loop on the DVCR 100 request of GUI description 202.

[0043] The GUI presentation engine 200 is utilized in the client device such as the DTV 102 to interpret GUI descriptions 202, 204 written in the HTML4.0 document description language and the associated specifications (below), and to create the graphical form for display to the user. The GUI presentation engine 200 includes the following e.g. attributes: 1)

window (GUI) minimum default size of e.g., HOx640 pixels (480x640 where 480 vertical, 640 horizontal). This default size is to insure the intended appearance in the GUIs 202, 204 is transferred to the user in the browser 200. The transferred GUIs 202, 204 are displayed in a window 480x640 pixels or magnified larger with the same aspect ratio unless otherwise directed by the user; 2) still image compression formats: e.g., GIF89a, JPEG, and PNG; 3) style sheet formats and fonts: e.g., CSS1 and CSS2; 4) fonts such as the following e.g. built-in fonts are required for the client device to free simple server appliances from having to support such fonts. Minimum one font from each generic Latin family can be selected: e.g., Times New Roman, from 'serif' family; Helvetica, from 'sans-serif' family; Zapf-Chancery, from 'cursive' family; Western from 'fantasy' family; and Courier from 'monospace' family. Other fonts can also be utilized; and 5) scripting language e.g., ECMA-262. Examples of the GUI presentation engine 200 include Web browsers such as Explorer™ and Netscape™ configured/customized as desired.

[0044] One or more of the server devices (e.g. a 1394WEB network, controlled appliance Web server such as the DVCR 110), include the following six enumerated components:

1) HTTP1.1 web server protocol, with section '8.1.2.1 Negotiation' of the HTTP1.1 specification regarding connection modified such that an HTTP1.1 server device (e.g. DVCR 110) assumes that a HTTP1.1 client device (e.g., DTV 102) intends to maintain a persistent connection with the server device. The persistent connection in the 1394WEB network 100 allows full status reporting from e.g. the server device DVCR 110 to the client device DTV 102 while the GUI 202 of the DVCR 110 remains visible in the browser 200 of the DTV 102. Further, a method using HTTP conditional GET to obtain the latest status of server devices can be used. Whenever the user returns to the home network directory or causes it to be refreshed, the browser 200 redisplays the page in its entirety. This is necessary because the HTML that underlies the home network directory may have been regenerated if a device has been added to or removed from the network 100. It is also possible for device icons to be updated to reflect changes in their device's operating state. As such, browsers implemented by EIA-775.1 devices utilize HTTP "conditional get" requests to determine whether or not fresh copies of web pages or graphics should be retrieved from the server.

2) Device home page GUI descriptions 202, 204 written e.g. in HTML4.0, include file e.g. icon.htm, name.htm, logo.htm, index.htm, gif files, etc.. The file index.htm is referenced by HTML links included in device icon.htm and name.htm HTML files, wherein index.htm can be optionally named e.g. AINDEX.HTML© or AINDEX.HTM©. File named INDEX.HTM is not required to be a standard name because the ICON.HTM and NAME.HTM are made with hyperlinks to the 'INDEX.HTM', therefore the name is arbitrary. ICON.HTM and LOGO.HTM reference the actual graphics files in the same device e.g. LOGO.GIF and ICON.GIF. The descriptions 202, 204 are accessible by the devices (e.g., HTTP devices) in the network 100. To guarantee a desired appearance, the control GUI design can be for a default GUI size of e.g. 480x640 pixels. For example, a transferred GUI 202 can be displayed in a window of 480x640 pixels in the browser 200 or magnified larger with the same aspect ratio unless otherwise directed by the user.

3) At least two device ICON files are provided to represent the device in a top-level network page 220 (FIGS. 5-6) in the browser 200 showing information about the devices connected to the network. An ICON can comprise a graphic file type (e.g. GIF, JPG or PNG) and named ICON.HTM. In one example, ICON.HTM(DVCR) references the INDEX.HTM file in the HTML page 202 and ICON.HTM(DTV) references the INDEX.HTM file in the HTML page 204. The top-level link for the control pages (e.g., INDEX.HTM) of the device can be ICON.HTM. The browser 200 places the icons and links therein) of a plurality of devices in the network 100 in the top-level HN directory page 220 for service discovery by the user. Then user clicks the ICON displayed in the page 220 and the device page (e.g. INDEX.HTM in page 202) is fetched. The default displayed HN directory is the top-level discovery page.
A number of additional and different graphic icons can also be utilized, for example, to represent device status, user configured preference or manufacturers formats which may be substituted for the icon graphic. In a discovery process described further below, ICONs from the devices connected to the network 100 are collected together and displayed in the top level network devices page 220 for selection by a user. An example device ICON specification comprises: File name ICON.HTM accessible by the HTTP server (files names are in a directory, file space, accessible by the web server so that they can be retrieved and forwarded over the network to the browser); Graphic file type such as GIF, JPG or PNG; and Icon graphic with a maximum size of 70(V)x130(H) pixels.

4) At least two device LOGO files are provided to represent the device in the top-level network devices page. LOGO can comprise a graphic file type (e.g., GIF, JPG or PNG) and named LOGO. HTM. In one example, LOGO.HTM (DVCR) references the INDEX.HTM in the HTML page 202 and LOGO.HTM(DTV) references the INDEX.HTM in the HTML page 204. In one version, the top-level link for the control pages (e.g., INDEX.HTM) of the device can be LOGO.HTM. All device logos are placed in the top-level HN directory page 220 for service discovery by the user. Then user clicks the LOGO displayed in the page 220 and the device page (e.g. 202) is fetched. A number of additional and different graphics for manufacturer services can be substituted for the logo graphic format. According to the discovery process, LOGOs from devices connected to the network 100 are collected together and displayed in the top level network devices page 220 for selection by a user. An example device LOGO specification comprises:

File name LOGO.HTM accessible by the HTTP server; Graphic file type such as GIF, JPG or PNG; and logo graphic maximum size of about 70(V)x130(H) pixels.

5) At least one device NAME is provided to represent the device in the top-level network devices page. NAME comprises TEXT in an HTML file NAME.HTM. This text can also reference control pages (e.g., 202). This is a top-level link in the discovery page to the control interface of the device. The text provides a way to distinguish identical devices whereby for e.g. two identical DTV's can be distinguished by adding NAME text 'Bedroom TV and 'Family Room TV. The text can comprise a few words to clearly represent the device type e.g. DVCR or DTV. According to the discovery process, NAMEs from devices connected to the network are accessed along with corresponding ICONs/LOGOs and displayed in the top level network devices page 220 under the ICON/LOGO. An example NAME specification comprises: File name NAME.HTM accessible by the HTTP server; Text unspecified, such as, with Font size 10, two lines of text can be displayed under the corresponding ICON/LOGO. Therefore, for example the space size for the NAME.HTM text can be 20 vertical by 130 horizontal to match the ICON/LOGO (70 vertical x 130 horizontal). As shown by example in FIGS. 5-8, the format of the top-level UI 220 can comprise a matrix of icons representing the functions of the networked devices to the user. The name representing the device (from name.htm) is placed under the icon (from Icon.htm) from the same device. Logo (from logo.htm) may be placed e.g. in any vacant icon position. As the Top-level description 250 (described further below in conjunction with FIGS. 9A-C) is generated independently by UI capable devices, the exact design need not be prearranged. The icon, logo and name maximum sizes can be prearranges to facilitate design of the GUI matrix.

6) A device information summary home page description document written in HTML4.0 can be provided, named e.g. "info.html" or "info.htm", and made accessible by the HTTP server for the discovery process. A link can be provided to INFO.HTM information via control pages e.g. 202, 204. The device information summary homepage provides the user a device summary instead of the detailed control interface as shown in the device homepage. Table 2 shows device attributes text that are included and others that can be included. This table can be extended to included other attributes.

<Table 2 - Device information summary>

Name	Value
Device Name	Device name (user configurable)
Device Location	Device location in home (user configurable)
Device Icon	Current Device ICON name
Device Type	Device type or category (VCR, DSS, TV, etc.)
Device Model	Device model
Manufacturer Name	Name of device manufacturer
Manufacturer Logo	Manufacturer Logo image name
Manufacturer URL	Device manufacturer's URL
Stream Source Name Default	Service: Default source device name for this Device's destination service
Stream Destination Name Default	Service: Default destination device name for this Device's source service
Stream Source Attributes	Type of service device can deliver (attributes and capability)
Stream Destination Attributes	Type of service device can receive (attributes and capability)

[0045] Table 2 includes device summary information such as Manufacturer Name, Manufacturer Logo image name, and can further include a Manufacturer URL for help if there is an available Internet connection to the manufacturers Web site. Table 2 can further include a user configurable Device Name and Device Location in the home. There can be several variations of the Device Icon representing different states of the device. The Device Icon attribute field includes the name of the current icon. Therefore, the device summary information page can provide immediate device state information to the user by displaying the icon representative of current state.

[0046] Each device can include one or more services, e.g. video Stream Source or video Stream Destination. Each source capability has a complementing Default Destination capability and each destination capability has a complementing Default Source capability. This Stream Default Name entry can be used e.g. to automatically default the nearest DTV to be the destination when a DVCR is being controlled as source to eliminate having to select the DTV each time.

EPNSDOCID: <EP_____1145244B1_L>

A background cross-referencing of the Stream Default Name to 1394 address is provided. The video stream services are provided by the 1394 interface itself (not by Web model). As such there is a linkage of the default source or sink to the 1394 address mechanism. The user can access a device and select a name for default, which is then saved on the device. The device's software agent must find the 1394 address and parameters for the 1394 s/w to enable the default stream when required.

[0047] Using the Source and Destination service attributes, new servers/services can be implemented while maintaining compatibility with existing host or device (nodes) and services. For example, if a new server device providing a new service is developed that is compatible with an existing server device, both the new and existing servers can be added to the attribute list of the new node while maintaining compatibility with existing nodes using the existing server in the network 100. The user can select a compatible device for purchase. These provide a user with "ABOUT" information to check capabilities of existing equipment e.g. prior to purchasing new equipment where compatibility is desired.

[0048] A discovery process for every device supporting the 1394WEB standard (e.g. devices capable of displaying a user interface) gathers device information from devices connected to the network 100 to generate the top-level user control page description for the home network, wherein each device is represented by a graphical icon reference and a textual name reference detailed above. The top-level description can include a default page for a presentation engine such as the browser 200, wherein the browser 200 collects the graphic images and names from the devices as it renders the network top-level graphical user interface 220 (GUI) displayed in the browser 200 as shown by example in FIGS. 5-6. The dynamically created top-level HN directory page 220 is made the default page for the browser (first page displayed when the browser is launched).

[0049] With reference to FIG. 4B, example operation steps include: 1) the browser 200 in device 102 is launched, 2) the browser 200 fetches and presents HN-Directory HTM (Top-Level UI) from the page 204, 3) the browser 200 fetches the HTM files Icon.htm and names.htm from pages 202, 204 and presents in the Top-Level UI, 4) the browser 200 fetches any graphics files (e.g., GIF) from pages 202, 204, and presents in Top-Level UI, 5) the browser 200 is then able to present the full HN_Directory page 220 (page 220 is made with hyperlinks to 'INDEX.HTM' files for different devices connected to the network 100), and 6) when a user clicks e.g. DVCR icon in GUI 220 to control the DVCR 110, a corresponding hyperlink in the top-level page 220 to 'INDEX.HTM' of the DVCR 110 is used to retrieve the'INDEX.HTM' (top control page of DVCR) from page 202 in the DVCR 110, and present the DVCR control page to the user (e.g., if the frame that was clicked (e.g. the Icon.htm frame) is not large enough, a graphic is presented in another copy of the browser with full frame size). The user can then command and control the DVCR 110 using the control interface provided by 'INDEX.HTM' of the DVCR device 110 presented by the browser 200 in the DTV 102

[0050] The name 'INDEX.HTM' is arbitrary because the ICON.HTM and NAME.HTM are made with hyperlinks to the 'INDEX.HTM'. However, ICON.HTM and LOGO.HTM reference the actual graphics files (e.g. LOGO.GIF and ICON.GIF) in the same devices. In one embodiment, LOGO.HTM can be optional if a logo for a device is optional. The HN_Directory HTML file can have a standard name so that it can be accessed from another device.

[0051] FIGS. 5-6 show that the host device, such as a client device (e.g., DTV 102, HDTV1) or server device (e.g., DVCR 110) that generates and presents the top-level GUI page 220 can assume priority and use a larger size icon for the host device's icon, name, logo, etc. in one version, only devices with servers (services on offer) are displayed in the GUI 220 (a "Client device" comprises device with Client capability, where if it is only client then it is not displayed in the top-level GUI as there is no service to offer). The discovery process reads information from the 1394 address space data storage (configuration ROM structure), as defined in clause 8 of ISO/IEC 13213. Although called 'ROM' it is assumed that the address space is write-able to allow user configuration and modification of user relevant stored values. The contents of the configuration ROM and the discovery process are described further below.

[0052] Device naming, addressing and discovery processes for home or local network control of consumer devices using Internet, Web and 1394 technology, can be different from the requirements and practice in the general Internet space. As such according to an aspect of the present invention for in home or local network control of consumer devices, special processes including device discovery, addressing and naming requirements are utilized. For example, the home network must fully function without the presence of external communications and services, without a network administrator, and configuration must be fully automatic. User control can be in many cases entirely keyboard-less. Further, the IEEE1394 protocol is utilized to provide a sophisticated interface including features that can be provide simple, efficient and superior discovery and configuration functions.

[0053] FIG. 7 shows a block diagram of a network 300 constructed in accordance with another embodiment of the present invention. To facilitate understanding, identical reference numerals have been used, where possible, to designate identical elements that are common throughout all the figures herein. As depicted in FIG. 7, a 1394 serial bus 114, described above, electronically connects multiple devices including server devices 14 (e.g., DVD 108, DVCR 110) and client devices 12 (e.g., DTV 102) on the network 100, described above in reference to FIG. 2, wherein the devices communicate using the example layered interface model of FIG. 3 as described above.

[0054] The network 300 is not restricted to using a 1394 serial bus, and, in alternative embodiments of the present invention, other bus types, such a Ethernet, ATM wireless, etc., can be used as the physical layer if they meet the

particular throughput requirements of an individual network (e.g., a home network) . As depicted in FIG. 7, the network 300 includes several devices connected to the 1394 serial bus 114. In this example, the devices include a DBSS 104 for receiving transmission signal from a satellite 122 for subsequent display. Associated with the DBSS is a network interface unit ("NIU") which, among other things, provides an interface between the DBSS satellite transmission and the 1394 serial bus 114. A digital video device (DVD) 108 is also connected to the exemplary network 300. The DVD 108 can be used to source digitally encoded videos for display on e.g. a digital television. Also connected to the exemplary network 100 is a digital video cassette recorder (DVCR) 110, a digital TV (DTV)102. In this example, the DTV 102 provides a human interface for the network 300 by employing browser technology to allow users to control and command for devices over the home network 300. A second DTV 103 provides another human interface for the network 100 by employing browser technology to allow users to control and command for devices over the home network 100. The DTVs 102 and 103 can provide human interfaces for the network 300 as each DTV comprises a screen for displaying HTML pages. However other devices having display capability can be used to provide human interfaces. Thus, in certain embodiments of the invention, a device such as a personal computer 105 (PC) is used to provide a human interface for a respective home network, as a PC 105 typically embodies a screen display unit.

[0055] The 1394 serial bus 114 is depicted as using the HTTP/IP interface protocol, and preferably HTTP/TCP/IP, wherein IP provides packet format (a one-way write only model), TCP provides an error free version of IP (e.g., ensures packets arrive and in correct order), and HTTP provides 2-way connection (packet to server will expect a response -a 'read' model). Certain devices can require other protocol interface types (e.g., TCP/IP, UPD/IP, FTP/IP, TELNET/IP, SNMP/IP, DNS/IP, SMTP/IP). In certain embodiments of the invention, a proxy 116 can be used to interface two networks using dissimilar interface protocols on their respective mediums which, when connected, comprise the network 300.

[0056] For example, as depicted in FIG. 7, the 1394 serial bus 114 using the HTTP/IP interface protocol is connected by a proxy 116 to the Home Automation network 118 (e.g., X10). By using the proxy 116 as HTML/HTTP/CTP/IP/1394 proxy for VCR-Commands/AVC/FCP/1394, to interface between HTML/HTTP/TCP/IP and X10 protocols, DVCR 120 is also accessible on the network 300.

[0057] In this embodiment, the network 300 can be connected to an external network 119 of dissimilar type (e.g., Ethernet) to the 1394 Serial bus, via a bus 121. A proxy 117 is used to interface the two dissimilar medium types. For communication between the addressing scheme of the external network 119, and the addressing scheme of the network 300, the bridge 117 comprises a Network Address Translation (NAT) boundary. This technique can be utilized for company LAN's and is a 'divide and conquer' approach to the complex problem of satisfying various network's differing IP address requirements and prevents 'running out of IPV4' addresses. The external network can include e.g. CABLE-TV network 115 via Ethernet to the telephone e.g. ADSL), providing broadband connection to the Internet and WWW. The Ethernet 119 provides the bridge function to the external network. The bridge 117 or Ethernet 119 may provide the NAT address conversion function. If the Ethernet is to provide local private (to home only) addressing (e.g. as defined by then IETF standard RFC 1918) then the NAT function is in the Ethernet 119. Existing cable modems are set up with a global address and also internet global address for the PC on the Ethernet (in this case the NAT is in the bridge 117).

[0058] The aforementioned device naming, addressing and discovery processes for the network 300 are now described. For device naming, point and click Web operation (e.g., using GUI/Web) does not require name services (DNS, Domain Name Service). The Web GUI provides an abstraction layer, and the addresses are hidden as hyper-text links invoked by user'clicks' to active GUI areas (e.g., buttons). Any change to the devices in the local network 300 causes the top-level discovery GUI page 200 (FIGS. 5-6) to be recreated by the browser 200 (FIGS. 4A-B) representing the status of the devices in the network 300 at that time and by default presented to the user for immediate use.

[0059] For device to device control a different look-up service is utilized for more than names (e.g., service look-up and application look-up). As such, DNS may not provide the necessary features for device to device control. However, a device (e.g., a 1394 connected PC) can access a DNS service as usual. DNS is not required for discovery or operation of devices/services within the home, but DNS (name to address) look-up service is required for external accesses e.g. from a PC. When a name e.g. "www.yahoo.com" is typed in to a Browser then look up take place for the IP address of the Yahoo computer, i.e. 216.32.74.52, because the Internet (even home Internet) operates with addresses.

[0060] For a 775WEB UI device which includes an agent for generating the HN top-level directory GUI description and also includes access to the special company web server e.g. homewideweb.com (IP address), can also have the DNS address knowledge. The DNS server computer IP address can be any IP address under the control of the manufacturer. Effectively the DNS address is built-in to the device (or can be updated if the agent is made to be update-able and is later updated).

[0061] For device addressing, in one embodiment of the invention, utilizing fixed IP addresses from a large address space can afford the simplest and most reliable network configuration, and the readily accessible ROM data space in the1394 interface allows utilization of fixed IP addresses therein. In another embodiment of the invention, non-fixed IP (dynamic) addresses can be utilized, wherein an abstraction layer (e.g., name or look-up mechanism) is employed to retain pre-organized communications.

[0062] For IP address configuration, the following protocols can be utilized: 1) Dynamic Host Configuration Protocol

(DHCP) with DHCP servers and DHCP clients, 2) DHCP clients resort to auto-configuration (DHCP server not present), and 3) preferably, FWHCP (Fire-Wire Host Configuration Protocol) server agent(s) and FWHCP clients, described further below. The auto-configuration in 2) above is that proposed as an IETF Draft "draft-ietf-dhc-ipv4-autoconfig-04.txt".

[0063] DHCP requires support of the BOOTP/UDP protocol, and replicates what is done within the 1394 specification and provides features such as lease time and dynamic addressing. Typical DHCP requires management by an administrator and must be configured and adapted to the network requirements of mass manufactured consumer electronics (CE) appliances where, for example, multiple identical CE appliances with DHCP server built-ins must be considered.

[0064] The 1394 technology provides 'Plug-in' or 'Power-up' reset and following 'Self-ID' sequences, well suited for network configuration. Further, the 1394 specification provides a built-in'ROM' address space well suited for storage of, and access to, configuration data (e.g., IP addresses). As such, in a preferred embodiment of the invention, an IP address configuration agent (FWHCP) and discovery page for user control of 1394 devices are utilized. FWHCP provides IP address configuration for 1394WEB and 1394 devices. The purpose and result of FWHCP is similar to DHCP (i.e., a server to identify and assign the local IP addresses), but in operation FWHCP uses data in 1394-address space and 1394 commands. FWHCP provides IP address configuration of 1394WEB devices on the 1394 network avoiding collisions with devices on adjacent attached networks other than 1394. Devices are manufactured with a built-in IP address from the 10.x.x.x range. In the unlikely event of a collision, FWHCP sets a new IP address and saves it in the device.

[0065] DHCP/Auto-configuration can be utilized for devices on networks other than 1394. DHCP protocol provides client "requested IP address". Preferably, the requested IP address space is selected from the upper part of the 24 bit RFC1918 range (10.128.1.1 to 10.254.254.254). By choosing part of the allowed private address range for 1394 IP addresses and another part for other configuration methods (e.g., DHCP and DHCP/Auto-Configuration) then compatible and non-interfering addresses are generated for a heterogeneous network and allow FWHCP and DHCP to coexist.

[0066] While choice of non-overlapping IP addresses for 1394 and adjacent networks is desirable, the heterogeneous network using FWHCP will configure successfully even if they do overlap. Also, DHCP clients check their assigned IP address with a test ARP message before using it. As such, different address configuration methods can coexist successfully.

[0067] Referring to FIG. 8, an example process according to the present invention for communication between a 1394 network (e.g., network 300) and a non-1394 network (e.g., Ethernet 119) for IP address configuration is described. In this case the 1394 network 300 utilizes FWHCP configuration and the non-1394 network 119 utilizes DHCP configuration or other method. Generally, 1394 devices (such as DTV and DVCR in FIG. 7) do not support DHCP. The 1394 DEVICE-3, for 1394 network to non-1394 network communication, includes an IP address in the 1394 ROM space and provides support for FWHCP for a 1394 device. The DEVICE-3 further includes means for supporting the configuration mechanisms on the non-1394 network, and maintains an extension data leaf in the 1394 ROM space for IP addresses of devices on the non-1394 network. As such, configuration processes (e.g., FWHCP for top-level UI description generation) on the 1394 network 300 can include use of IP addresses on the non-1394 network by selecting IP addresses from the extension data leaf. The non-1394 network configuration operates to provide the IP addresses for the 1394 extension data leaf.

[0068] According to the discovery process (agent), 1394 specification 'plug-in' reset and self-ID is utilized for configuration and can be used for IP address configuration. Preferably, fixed IP addressing is utilized for home networks, however dynamic IP addressing can also be utilized. DNS is not required within 1394WEB control because a top-level GUI description is created with hypertext-links that use IP addresses rather than names. Preferably, the IP configuration agent (FWHCP) for the 1394 network is utilized for IP configuration using 1394 ROM data and 1394 commands, however DHCP can also be utilized. FWHCP utilizes lower half of RFC1918 10.LH.X.X addresses and other home networks (not 1394) use upper half 10.UH.X.X. Preferably, the FWHCP server agent is built-in to any device that can be a client (control initiator). Where there are several client devices connected to the 1394 network, only the client device with the highest Global Unique Identification (GUID) operates. GUID comprises a number built-in to the interface. If there are multiple FWHCP agents available on the 1394WEB network then there is an initial self-election process to determine the one that will operate and all others remain quiet. The highest GUID will operate. In other versions, highest bit-reversed-GUID can be used.

[0069] A device interfacing to a non-1394 network supports a ROM extension leaf of IP addresses on the non-1394 network. This allows inclusion of the IP addresses on the non-1394 network in the 1394 top-level GUIs (e.g., FIGS. 4A-B, GUIs 202, 204). Control data bits in the 1394 ROM space are used to control the operation of three configuration agents: 1) 1394 Self_ID count, 2) IP configuration FWHCP, and 3) UI description generation described further below.

[0070] Initially 1394 Self-ID count discovers the existence of devices. After a bus reset (caused by power up/down or device attachment/detachment) 1394 software in the device observes the automatic configuration process (1394 self-ID cycles) for the purpose of counting the number devices. This is a normal part of 1394 software for any 1394 device. Then, IP Configuration FWHCP (the one self-elected FWHCP) probes the discovered devices and checks their built-in IP address. Discovered duplicate (colliding) IP addresses are disabled and a new address is assigned to the device. Then, UI description generation agent (UI or other devices), reads all 1394WEB device IP addresses and generates a top-level device directory Graphic User Interface file in HTML of top-level icon pages from each device later rendered

by a Web browser for User discovery of devices for control.

[0071] According to the present invention each device in the 1394 network 400 can generate its own top-level network UI description 250 (FIG. 9C). The UI description 250 is used by a presentation engine such as the browser 200 in a client device to generate and display a top level directory page such as page 220 in FIGS. 5-6. After the 1394 Self-ID agent has enumerated all devices connected to the 1394 network 300, the top-level UI description 250 is generated separately by all UI devices (and non-UI devices as desired). A device (e.g., DTV) can select a more prominent (e.g., larger) icon to represent that device, and make the entire GUI 220 with a different look. This technique provides substantially more reliable operation than a centrally generated GUI for operation of all device, because each device can generate its own UI description 250 and display a GUI (e.g., top level page 220) based thereon without dependence on another device. In each UI description 250, device icon and logo image files of the devices currently connected to the network 300 are referenced by icon and logo HTML 'pages' and name text wrapped in an HTML page (ICON.'Graphic' referenced ICON.HTM is in pages 202 and 204 which also include the control pages for the device; Fig. 5 below also shows the ICON.HTM, LOGO.HTM and NAME.HTM in a top-level directory page). HTML frames are used to create the top-level directory UI description 250 for network devices in each network device as desired.

[0072] As such, advantageously, a useful layer of abstraction is provided to allow use of alternative file names and types for e.g. identification graphics in the network devices without need for change in the top-level description 250 generated in each device. The name text is also placed in an HTML description 202, 204 (NAME.HTM is in pages 202, 204), allowing a user to configure the name text at a device e.g. DTV to change to e.g., DTV-BED2 through one of the device GUI pages 220. As such, the page 220 is displayed as the Browser is launched after a reset. The user sees and clicks DVCR ICON graphic, whereby DVCR top level control GUI 202 is fetched (with 'Play' button etc.). User clicks one of the buttons e.g. "Configure Device NAME" which is another GUI (of hierarchy of control pages for DVCR) with a large selection of different names. User clicks one name out of the lists of names provided e.g. "Master Bedroom DVCR". Software on the device changes the file names so that the file named NAME.HTM contains the text "Master Bedroom DVCR" (the old default NAME.HTM file that contained DVCR is changed to some other name).

[0073] Appearance of the GUI 220 is more stable in the event of 'bad citizen' devices having too much or oversized text or oversized logos. In this case the frames isolate the problem and prevent the bad items from adversely affecting the appearance of the entire top-level GUI 220.

[0074] Referring to FIGS. 9A-C, 10, 11 example functional blocks and connections to data and control bits and flowchart of an embodiment of a system architecture 400 for the aforementioned discovery process are shown. The system 400 comprises five primary elements: 1)1394 nonvolatile memory space (IEEE1212R ROM) 402 for configuration data and control data bit storage; 2)1394 Device Discovery Agent (1394DDA) 404; 3) IP Address Configuration Agent (FWHCP) 406; 4) UI Description Generation Agent 408; and 5) GUI Generation and run-time environment 410 (e.g., Web Browser 200 in FIG. 2). Further, FIG. 10 shows an example flow diagram for the DDA and FWHCP agents in system 400 operating in connection with the functional blocks in FIGS. 9A-C. And, FIG. 10 shows an example flow diagram for the UIDGA agent in system 400 operating in connection with the functional blocks in FIGS. 9A-C.

[0075] Referring to FIGS. 9A and 10 all devices include the 1394 device discovery agent (1394DDA) 404 to enumerate the devices on the 1394 bus, after a reset, and to write the value into the local 1394 ROM space 402 for communicating the value to other functional agents (steps 500, 502). For synchronizing (inhibiting) commencement of other configuration agents, the 1394DDA agent 404 also sets the 'configuration operating' control bits. The discovery agent/mechanism can use means, other than the ROM space, to communicate information between the configuration agents that are local to one device and where the information does not need to be seen by other devices.

[0076] All devices in the network 300 include the following information relevant to the discovery and IP address agents 404 and 406, respectively, for the1394WEB in the 1394 configuration ROM 402: 1) Built-in 64 bit GUID (Global Unique ID, in 1394 specification); 2) Built-in IP address from the RFC 1918 private address space in the range '10.1.1.1' to '10.127.254.254'. Manufacturers can select a value from the GUID such that chance of collision is minimized. The upper portion of the private address space (i.e., 10.128.1.1 to 10.254.254.254) is reserved for devices on bridged networks; 3) Assigned IP address in the range '10.1.1.1' to '10.127.254.254' (assigned by operating FWHCP agent 406); 4) IP address extension leaf for IP devices on bridged networks; 5) Assigned Count of 1394 devices (assigned by 1394DDA agent 404); 6) Control/status bits to indicate Configuration-In-Progress Synchronization control for 1394 Device Discovery Agent 404, and to indicate IP-Address configuration (The control bits indicate the configuration is in progress and therefore the values, in ROM data other than the control bits, for 1394DDA and IP address are not checked or not written and therefore should not be used). The bits further indicate which IP address is valid (assigned or built-in), and whether an FWHCP server agent 406 is present in the device; 7) HTTP web server to allow files in the device's file space to be accessed remotely; and 8) device information 202, 204 including actual 'icon', 'name' and 'logo' HTML files and other referenced graphic files accessible through the Web Server. The above summarized information is detailed in the 1394 ROM space description below.

[0077] The content of the general 1394ROM structure 402 is specified in IEEE1212r, IEEE1212 and IEC61883. The ROM structure 402 is a hierarchy of information blocks, wherein the blocks higher in the hierarchy point to the blocks

beneath them. The location of the initial blocks is fixed while other entries are vendor dependent, but can be specified by entries within the higher blocks.

[0078] Table 3 shows the Bus_ Info_Block and Root_Directory of the configuration ROM 402. The first byte of each entry is known as a key and identifies the type of entry. The following can be implemented in the configuration ROM of all devices making use of the EIA-775 specifications, including display devices such as DTVs and source devices such as DVCRs, STBs, etc. There may be several other structures required based on other protocols to which each device conforms. Table 3 includes information for a device which also complies with the IEC61883 protocol. The Root_directory contains pointers to a Model_Directory and three Unit_Directory entries (IEC61883, EIA-775 and 1394WEB), to indicate that the device supports EIA-775 as well as 1394WEB protocols. The Root directory entries are useful to other 1394 devices to discover the protocols and software (also called services) supported by this 1394 device.

<Table 3 - Configuration ROM>

Offset (Base address FFFF F000 0000) Bus_Info_block					
Offset					
04 00_{16}	04	crc_length	rom_crc_value		
04 04_{16}	"1394"				
04 08_{16}	flags	reserved	cyc_clk_acc	max_rec	reserved
04 $0C_{16}$	node_vendor_Id				chip_id_hl
04 10_{16}	chip_id_lo				

[0079] Wherein, 04 $0C_{16}$ and 04 10_{16} are also known as the 64 bit GUID or Global Unique ID.

Root_directory			
Offset			
04 14_{16}	root_length		CRC
	03_{16}	Model_vendor_ Id	
	81_{16}	Vendor_name_textual_descriptor offset	
	$0C_{16}$	Node_capabilities	
	$8D_{16}$	Node_unique_Id offset	
	$D1_{16}$	Unit_Directory offset (IEC 61883)	
	$D1_{16}$	Unit_Directory offset (EIA-775)	
	$D1_{16}$	Unit_Directory offset (1394WEB)	
	Optional		
xx xx_{16}	$C3_{16}$	Model_Directory offset	

[0080] The IEC_61883 unit directory is shown in Table 4. This directory is referenced by the Unit_Directory offset, in the Root Directory (i.e., Table 3.). In the Unit_SW_Version field, the least significant bit specifies AV/C (0) as specified in IEC 61883.

<Table 4 - IEC_61883 Unit Directory>

Unit_Directory (IEC 61883)	
Directory length	CRC
12_{16}	Unit_Spec_ID (1394TA = 00 A0 $2D_{16}$)
13_{16}	Unit_SW_Version (first pass key = 01_{16})
....	<<possibly other fields>>

(continued)

Unit_Directory (IEC 61883)	
Directory length	CRC
....	.....

[0081] The EIA-775 Unit Directory is shown in Table 5. The following EIA-775 specific information appears in the EIA-775 Unit Directory.

<Table 5 - EIA-775 Unit Directory>

directory length	CRC
12_{16}	**Unit_specification_ID** (EIA-775 = 005068_{16})
13_{16}	**Unit_software_version** (010100_{16})
....	<<possibly other fields>>
.....	.....

[0082] The Unit specification_ID specifies the identity of the organization responsible for the architectural interface of the device and the specification. In this example case, the directory and identity value=005068_{16} refers to the EIA as the responsible body and the EIA-775 control architecture specification.

[0083] The Unit_software_version designates EIA-775 revision level supported by the device. The format is shown in Table 6.

<Table 6 - **Unit_software_version** coding>

First octet	01_{16}
Second octet	Major Version Number (currently 01_{16})
Third octet	Minor Version Number (currently 00_{16})

[0084] The 1394WEB Unit Directory is shown in Table 7. The following 1394WEB specific information appears in the 1394WEB Unit Directory.

<Table 7- 1394WEB Unit Directory>

directory length	CRC
12_{16}	**Unit_specification_ID** (1394WEB = $00XXXX_{16}$)
13_{16}	**Unit_software_version** (010100_{16})
38_{16}	**Discovery_control_bits**
39_{16}	**Assigned_Count_of_1394_devices**
$3A_{16}$	**IP_Address_Built_In**
$3B_{16}$	**IP_Address_Assigned**
	IP Address_Extension Leaf
$-_{16}$	<<possibly other fields>>

[0085] The Unit_Specification_ID specifies the identity of the organization responsible for the architectural interface of the unit and the specification. In this example case the directory and identity value=$00XXXX_{16}$ refers to the responsible body and the 1394WEB control architecture specification.

[0086] The Unit_software_version designates the 1394WEB revision level supported by the device. The format is shown in Table 8.

<Table 8 - Unit_software_version coding>

First octet	01_{16}
Second octet	Major Version Number (currently 01_{16})
Third octet	Minor Version Number (currently 00_{16})

[0087] Key value (38_{16}) permitted by the IEEE1212R specification section 8.8 for the private use by the owner of the directory and architecture is used for the Discovery_control_bits immediate value.

<Table 9 - Discovery_control_bits>

				FWHCP Server Agent	Configuration operating. Do not use (if True)		Which IP address?
X				Yes=1	1394 Dev. Count	IP-Address	Assignd_1B uilt-in_0
31	6	5	4	3	2	1	0 (LSB)

[0088] These are control bits in 1394 ROM space 402 accessible by local and remote device. The control bits are used by the IP address configuration agent 408 and the User Interface description generation agent 408 as described further below.

[0089] In one embodiment of the invention, said control bits provide the following information:

[0090] Bit 0 - Which IP address - Indicates which IP address is used or is in-use i.e, the Built-In address (=FALSE) or Assigned Address (=TRUE). This is set by the operating IP configuration agent FWHCP 406.

[0091] Bits 1, 2 - Configuration Operating Do not use - When set indicate that the 1394 device discovery and also, seperately, the IP configuration agents 404 and 406, respectively, are operating and therefore the values referred to are invalid as they can change or are notyet written. These bits are set by the local (device) 1394DDA agent 404. The1394DDA agent 404 clears the 1394 Dev. Count bit and the operating FWHCP agent 406 clears the IP-address bit.

[0092] Bit 3 - Presence of FWHCP Server Agent 406 B is set if the device has an operable FWHCP agent 406. This bit and GUID are used by the FWHCP agents 408 to determine which FWHCP agent 408 will operate

[0093] Assigned_Count_of_1394_devices (39_{16}) - Assigned immediate value of the count of 1394 devices in the network 300. The count is made as the 1394 interface goes though its self-ID cycles. The 1394 device discovery agent 404 generates the value, which is saved in ROM space 403 for subsequent use by the IP and UI configuration agents 406 and 408, respectively.

[0094] IP _Address_Built_In ($3A_{16}$) B Assigned Immediate Value. This address is assigned at manufacture time and built-in to the device. If this Built-in address cannot be used, an alternative address can be saved in the Assigned address space and the control bit set to indicate such.

[0095] IP_Address_Assigned ($3B_{16}$) - Assigned Immediate Value. If identical IP addresses are detected, the IP address configuration agent FWHCP 406 assigns this address to prevent collision. Further, the control bit is set to indicate such.

[0096] IP_Address_Extension Leaf_for_attached_network (BC_{16})- This directory entry is for the address offset to the data leaf for the IP address extension table, see Table 10. The data leaf contains IP addresses for devices on connected non-1394 networks (but also could be bridged 1394 networks). The table is included in communications devices of types (e.g., bridge) that connect through to foreign (non-1394) networks. The table can be expanded to include as many IP addresses as required. The address of the communications device itself should not be included in the table.

<Table 10 - IP_Address_Extension Leaf>

Leaf Length -1 (n)$_{16}$	CRC-16 $_{16}$
IP Address 1	

IP Address n	

[0097] In regards to Control word for Discovery Control Bits, use of a ROM entry for the actual Discovery Control Bits

word as defined herein works but is an example implementation. As ROM is not designed to be written efficiently (i.e., ROM areas have to be erased and writing them is slow relative to other hardware e.g. register).

Registers are provided in the 1394 hardware for data that must be written to frequently. In another version, a 1394 Register can be used for the 'Discovery_control_bits' control word. Registers are in a space also addressable by other devices, whereby another device can look up in the ROM the address of the Register and then write to that Register.

[0098] Referring Figure 9B, one or more devices include an IP address configuration agent (FWHCP) 406 (e.g., all UI devices and Gateway devices and any other device that can be a Control initiator). The FWHCP configuration agent 406 accesses all devices' IP address values in data in the1394 ROM 402 across the 1394 network 300. For synchronization commencement and completion of commencement of other applications (e.g., the UI description generation), the FWHCP agent 406 also accesses the 'configuration operating' control bits.

[0099] Referring to Figure 9C, devices capable of displaying user interfaces, and also some other devices (e.g., Gateway devices), can include the UI description generation agent 408 for generating the top-level UI description 250 in e.g. HTML. Because as detailed above only one IP configuration agent 406 operates per network 300, not all devices need to include the IP configuration agent 406, though all devices can include an IP configuration agent 406. If a device has the operating IP Configuration Agent 406 and is a User Interface Device then the IP configuration agent should operate before the UI Description Generation agent. The UI description generation agent (UIDGA) 408 utilizes information including control bits defined in the1394 ROM space 402 and other information (e.g., for determining which FWHCP operates is the Global Unique ID (GUID) of Bus_Info_Block of Table 3) for determining which IP configuration agent 406 (if multiple in the network) operates, synchronizing commencement and for access to the in-use IP addresses. Any device may have and operate a UIDGA for making the HN_Directory page (top-level discovery page). After the IP addresses are configured UIDGA reads the addresses to make the HN_Directory page. In each client device, when UI description generation is complete, the GUI generation and run-time environment 410 (e.g., Web Browser 200 in FIG. 2) uses the UI description HTML file 250 to access all devices HTTP file space for icons, names and logos (Icon.HTM, Name.HTM and Logo.HTM are contained in pages 204, and 204) to generate the full top-level GUI 220 for display in that client device. Web Browser uses HTML file 250 to render the actual GUI graphics, in the process accessing files from the devices e.g. Icon.HTM, Name.HTM and Logo.HTM and in turn accessing any additional files these files reference e.g. ICON.GIF and LOGO.GIF.

[0100] Referring to FIGS. 9A-C, 10 as discussed, each 1394WEB device in the network 300 can include the device discovery agent 404. The device discovery agent 404 enumerates the 1394 devices in 1394 address space connected to the 1394 bus, wherein the raw discovery is performed in 1394 hardware. The Self_ID and Physical Node Number Assignment and the steps leading to it is the basic discovery process performed by the Interface hardware/firmware. All devices monitor the Self_ID cycles and make a note of the existence of 1394 devices. This is a part of 1394 software for any 1394 device: 1) Reset -Bus reset propagates to all interfaces, on device power-up, device attachment and device detachment, 2) Tree Identification -Transforms a simple net topology into a tree, to establish a ROOT which is master for certain functions: Bus Cycle Master, Highest priority in arbitration for bus time, 3) Self Identification -Assigns Physical Node number (address) and also exchange speed capabilities with neighbors. Highest numbered node with both Contender Bit and Link-on Bit is Isochronous Resource Manager.

[0101] The discovery agent 404 writes the final count value of the devices to the 1394 ROM space to communicate it to other agents. The device discovery agent 404 is the first software agent to execute after a 1394 reset cycle, and control bits (Discovery Control Bits 2 and 1, Configuration Operating: 1394DDA, and IP_Address) are used to delay other agents, including the configuration agents 406 and 408, from execution until the discovery agent 404 has finished execution.

[0102] In one embodiment, the1394DDA agent 404 in each device performs the steps 500, 502 including: 1) setting synchronization control bits (i.e.,'1394DDA in progress' and 'IP configuration in progress' bits) in the device's own 1394 ROM space 402 to indicate that the 1394DDA in progress and IP configuration is in progress (IP configuration will not be in progress if 1394 DDA is executing) and that the values of 1394 device count and IP address are not valid, whereby said control bits inhibit other agents (e.g., 408) from operating prematurely; as such the 1394 DDA executes, then an elected FWHCP executes, and then (usually for UI device) UIDGA executes; 2) counting the number of 1394 self-identity sequences after a 1394 Reset to discover the number of devices and effectively their local node addresses for use by the other agents 406, 408; 3) writing the device count value to the device's own 1394 ROM space 402; and 4) clearing (e.g., to false) the synchronization control bit for '1394DDA in progress' in the device's own 1394 ROM 402, wherein the 'IP configuration in progress' bit remains set and is cleared later by the operating FWHCP agent 406.

[0103] Alternative Architecture for Configuration with IP Address list In network communication (bridge) device is possible. For example, the IP address list of IP addresses of devices on a bridged (e.g., non-1394 network) can alternatively be examined at the IP configuration stage by the FWHCP agent 406 rather than only at the UIDGA stage by the UIDGA agent 408. This allows the FWHCP agent 406 to detect and correct address collisions and therefore allow operation without having two separately defined address ranges, one for the 1394 network 300 and one for the non-1394 network 119. Correction of address collision can be accomplished by modifying the address of a colliding 1394

EPNSDOCID: <EP_____1145244B1_I_>

device as the bridged network IP address list cannot be modified by the aforementioned agents 406, 408 for the 1394 network 300. Configuration is more reliable if the FWHCP agent 406 can check the addresses in the bridged network 119 for collision prior to allowing the addresses used on the 1394 network 300.

[0104] Referring to FIGS 9A-C, 10 the IP Address Configuration software agent (FWHCP) 406, operates to provide 'Fixed' IP address management and to detect and correct IP address clashes in the mass manufactured 1394 devices. All 1394WEB UI devices include, and other devices can include, an FWHCP agent 406. Only one FWHCP agent 406 operates in the network however. The 1394DDA 404 agent is the first software agent to execute after a 1394 reset cycle, and as aforementioned the1394DDA 404 agent sets the '1394DDA in progress' and 'IP configuration in progress' bits to delay the FWHCP agent 406 until the 1394DDA agent 404 has executed to completion.

[0105] In one embodiment, the IP Address configuration agent 406 in a device performs steps including polling the 1394DDA configuration operation control bit (i.e., the '1394DDA in progress' bit) to determine if the 1394DDA configuration software agent 404 has executed to completion. If so, then the FWHCP agent 406 uses the count of devices determined by the 1394 DDA agent 404, and reads GUID's and Control Words from every device (step 504) to determine which device in the network 300 is selected to execute its FWHCP agent 406 (step 506). The selected device is one with an FWHCP agent 406 that finds it has the highest GUID (step 508). All other FWHCP agents 406 in other devices remain dormant (step 510). The operating FWHCP agent 406 reads the 'in-use' (active) IP address (determined by Discovery_control_bits BIT 0) from each local node (e.g. units present on the interface, host) and listed (step 512). In one version, the software agent makes a list for saving the IP addresses to an 'Array' as they are read (steps 514-518). The list will be in memory (RAM or DRAM) under the control of the compiler and OS. In-use status is determined by a bit setting in the device, which indicates whether the built-in or assigned address is in-use. In Table 7 the IP_address_assigned and IP_address_built_in are in the 1394Web Unit Directory.

[0106] The operating FWHCP agent 406 examines said list for collision among IP addresses listed therein (other collision detection and resolution methods can also be used) (steps 520-522). If a collision is detected, the FWHCP agent alters the colliding addresses by e.g. substituting the least significant 6 bits of IP address for their 6 bit node address (step 524). Only the minimum number of alterations are performed to relieve the collision. If one of the colliding addresses is already an assigned address, then that address is altered in preference to the colliding built-in address by e.g. incrementing the 6 bit substitute value and re-checking until the collision is resolved. The FWHCP agent 406 writes the altered value back to the device and the control bit (Discovery_Control_Bits: Bit 0) is set to indicate that the assigned IP address is in-use, and the built-in default is no longer in-use (step 526). The process is repeated for each IP address (step 528). After the collision resolution process, the operating FWHCP agent 406 accesses each device in turn and sets the 'IP configuration in-progress' bits in each device to e.g. 'false' to indicate that the indicated IP address is valid.

[0107] In conventional WWW operation, users access the same top level page. Referring to FIGS. 4B, 7 and 9-11, according to an aspect of the present invention however, all UI devices (e.g., devices capable of displaying user interfaces) include an UI description generation agent (UIDGA) 408 to independently generate a top-level UI page 220 for control of the devices on the local network (e.g., network 100, network 300, etc.) by users. In one example, a client device (e.g., PC) dynamically generates a locally saved default page 220 for user control of devices connected to the network 100. This allows each UI device (e.g., DTV 102) to generate a different view 220 of the home network e.g. with a larger more prominent icon for that UI' s devices displayed. As such, the user is readily made aware of which UI device is 'right here' (in front of the user) or in the case of access external to the home, no device is 'right here'. A device without a UI can generate a UI for another device but is unaware of type of device (e.g., Cable Modem generates UI of HN devices for user external to the home). In this case the actual UI device is unknown. Therefore no particular device is prominent in the GUI. Further, manufacturers of devices connected to the network 100 can provide their own GUI design 202, 204 in each device as desired. In addition later, Improved Browser and Web technology designs need not be hampered by existing technology.

[0108] Non-UI devices, particularly those devices performing a gateway function, can also include a UI Description Generation agent 408 to generate top-level GUI descriptions 250, without including GUI Generation and Run-Time processes 410 (e.g., Web Browser 200) to generate and display GUIs 220. With appropriate address use (e.g., using the RFC1918 private addresses on the local HN), this allows external WWW access to the 1394WEB network devices. External addresses are assigned 'real' IP addresses suitable for Internet use. Generally there is a unit (e.g., gateway type unit) with the UIDGA 408 which represents the home to the outside Internet The gateway's UIDGA generates a different UI description for the outside use (remote access case different from inside local device use), using the home's IP address with extended links to identify which home device local private IP address.

[0109] UI devices execute the following software processes to generate and display views 220 of the network 100/300; 1) 1394 Device Discovery Agent 404 described above, 2) UI Description Generation Agent (UIDGA) 408, and 3) GUI Generation and Run- Time (e.g., Web Browser 200) process 410. Referring to FIG. 11, In one embodiment, a UIDGA agent 408 in a device performs steps including polling the IP address configuration bits in the device's own 1394 ROM 402 to ensure completion of the FWHCP agent 406, prior to accessing any further IP information (step 600). Upon completion of FWHCP agent 406, using the count of devices generated by the 1394DDA agent 404, the UIDGA agent

408 then accesses the control word in each device currently connected to the network, to determine the settings for the 'configuration operating' false, and 'in-use' IP addresses bits (the UIDGA agent 408 makes the top-level HTML page, HN_Directory page, 220 shown by e.g., in FIGS. 5-6). Thereafter, the UIDGA agent 408 reads the actual in-use IP address value, and builds a complete list of the IP addresses of the devices currently connected to the network 300. The IP address list includes information (e.g., Icon, Logo, Name, etc.) from every device, and is written in HTML by using the IP address of each device. Before it can include the addresses, the UIDGA 408 finds the address of each device by accessing each device and checking to see which address is in use by reading Table 9, Discovery_control_bit, control bit (Bit 0). Then UIDGA 408 reads Table 7 Address either Built_in or Assigned. For devices that communicate to bridged networks, as determined by the presence of the extension IP address list entry in that device's 1394 ROM 402, the UIDGA agent 408 reads the extension IP-addresses from the list (IP_Address_Extension_Leaf) to allow those devices to be included in the GUI 220. The entry BC (IP_Address_Extension_Leaf) contains a reference link address that points to the actual data leaf. Devices on the attached bridged network are only included in the IP_Address_Extension_Leaf list if they also support the 1394WEB type of service i.e. they have Web Server and Icon.HTM etc and Control pages ('index.htm).

[0110] The UIDGA agent 408 reads the IP address list (step 602) and generates the top-level network UI description 250 (FIG. 9C) in e.g. HTML (e.g., Appendix 1) using the IP address list (UIDGA outputs the HN_Directory, top-level network UI page, HTML file) (step 604). The UIDGA agent 408 uses the IP Addresses in the hypertext links to each device for the Icon.htm, name.htm and logo.htm files. UIDGA writes an HTML file including the references to each discovered device's HTML page i.e. ICON.HTM, NAME.HTM, LOGO.HTM (e.g., Appendix 2, 3, 4). The UIDGA agent 408 then uses HTML files to reference items including the icon and logo graphics files and name data, rather than including the raw Icon.gif or logo.gif and raw name text in the top level UI description 250 (step 606). This allows said items to be changed by the corresponding device to reflect current status, customized by the manufacturer or configured by the user at the device, without causing any change in the top-level HTML UI description 250 in the controlling UI device. Though one graphic per device is shown in the example GUI pages 220 (FIGS. 5-6), customization allows inclusion of more than one graphic file referenced by ICON.HTM or LOGO.HTM and more text in the NAME.HTM. In one embodiment, HTML frames are utilized to implement the UI description 250 as showing in examples further below. Use of frames stabilizes the appearance of the GUI 220 in the event of 'bad citizen' devices. For example a device presenting too many words or overly large text in its 'name' frame will only affect that device's GUI look (by having some of the words truncated and not displayed) and not adversely affect the appearance of the whole Top-level GUI 220 in the UI device. The UIDGA then invokes the GUI generation process 410 (e.g., browser) in a client device to generate and display a user interface (step 608).

[0111] The GUI generation process 410 (e.g., Web Browser 200) utilizes the UI description 250 in e.g. HTML to generate GUI pages 220 on UI devices. In one example, to provide keyboard-less operation for consumer electronics devices (e.g., DTV) the Browser 200 at start-up defaults to reading and rendering a locally generated 'top-level-devices.html' description 250 to generate the network top-level control GUI 220. Locally as used here means in the same device (a UI device having a UIDGA that generates the device's own HN Directory (top-level) GUI of the network devices). HN Directory, Top level Network UI and Discovery page are the same. For personal computers (PC) with keyboard this need not be the default. For CE devices, launch of the Browser 200 is delayed until after completion of the UIDGA default page 250 generation by the UIDGA agent 408. In the event that UIDGA agent 408 cannot complete its tasks, then the Browser 200 displays an alternative UI page 220 showing a network configuration error occurred (e.g., "Unable to generate the HN_Directory Page because of xxxxxx. Try disconnecting device xxxxxx. Network configuration error number xxxxxx occurred. Contact service Tel service xxx-xxx-xxxx or Web service http://www.service.com.")

[0112] To generate the GUI 220, the Browser 200 fetches the 'icon.htm', 'name.htm' and 'logo.htm' files from device information 202, 204 in each referenced device (i.e., in the UI description, where for example ICON.HTM is in the HN_Directory Page HTML file) as defined by the HTML UI description 250. The contents of these pages 202, 204 (e.g. the icon graphic) need not be static and can be altered dynamically to reflect device status change, or after user customization. In order to display the most current top-level page 220, the Browser 200 does not cache the 'icon.htm', 'name.htm' and 'logo.htm' files. In another version, a check is always made first to determine if the device has made any changes to the HTML files 202, 204 it holds. HTTP "Conditional get" is used for checking the status of controlled device. Depending on the status code returned, the Browser 200 will either read from its cache or fetch a fresh or updated copy the HTML file 202, 204 from the devices. The HWW GUI display is not affected unless there is any change of the status of the controlled device.

[0113] The browser 200 does not attempt to display the top-level HN directory until it has been completely generated. If the HTML 250 is not generated within some reasonable amount of time, the browser displays an alternate page. If a network configuration error is the source of the problem, the alternate page might provide some technical support or user diagnostic assistance.

[0114] Whenever the user returns to the top-level HN directory or causes it to be refreshed, the browser 200 redisplays the page 220 in its entirety. This is necessary because the HTML 250 that underlies the top-level HN directory may have

been regenerated if a device has been added to or removed from the network 100. It is also possible for device icons to be updated to reflect changes in their device's operating state. As such, browsers implemented by EIA-775.1 devices use HTTP "conditional get" requests to determine whether or not fresh copies of web pages or graphics are retrieved from the server.

[0115] In this aspect, the present invention provides a User Interface description where user discovery of devices is thus made entirely with references (i.e. in the abstract), where the references are 'containers' for the discovery information (e.g., text and/or graphics) of each device and resident on each device. Each 'container' includes actual textual information and/or references to one or more graphics formatted information files where each file may include one or more images and/or text. Use of the reference 'containers' allows each device to choose its preferred UI content or graphics format or alter its UI content to be displayed (by changing the text or graphic information referred to) without need to have the UI description page altered in any way. Therefore, communication of changes with the generating agent software of the Discovery UI description is not required. In one version, devices reference their e.g. ICON and LOGO graphics files indirectly using HTML files enabled by creating the network Top-level description using HTML frames. Similarly the device name that is displayed under the icon is represented by NAME HTML file. HTML files are used to reference e.g. the Icon and logo graphics files and name data rather than include the raw Icon.gif or logo.gif and raw name text. This allows the item to be changed to reflect current status, customized by the manufacturer or user configured at the device without causing any change in the top-level HTML description. This level of abstraction allows the Top-level UI description to be always the same regardless of the graphics ICON and LOGO file names and types and NAME text to be displayed. Also the device may use different, multiple or dynamically change the graphics files and text displayed in the Top-level GUI without the change needing to be communicated to the UIDGA. The change is automatically included whenever the GUI is redisplayed. Use of frames also stabilizes the GUI display in the event of bad citizen devices using non-displayable graphics or text as the error is confined to the particular frame and doesn't affect the whole GUI. The change is automatically included whenever the GUI is redisplayed.

[0116] In one example, network devices top-level UI description is generated independently by any network device and certainly by devices capable of displaying UI (UI device). Generating a user interface in each device rather than generating a centrally UI, allows a device to show its own device Icon/text preferentially in the GUI. In addition each GUI is manufacturer customizable, user configurable and also more reliable because it does not depend on another device e.g. a single central server. This is demonstrated with the 1394 scheme above. Multiple UI generation is enabled because all device IP addresses are accessible via the 1394 interface. UI devices (with Browser) include UIDGA agent to generate their own top-level GUI description after a 1394 reset cycle when a device attached or power-up.

[0117] All UI devices independently generate a top-level UI page for control for the local network. This is different from the conventional WWW operation wherein users access the same top level page. According to one version the present invention, the client device (e.g., PC) dynamically generates a locally saved default page file for any purpose, allowing each UI device (e.g., DTV) to generate a different view of the home network e.g. with a larger more prominent icon for its own display. Further manufacturers have scope to make their own GUI design better then another. In addition later, improved Browser and Web technology designs need not be hampered by earlier technology.

[0118] Referring to Appendices 1-4, illustrative examples for the following are provided: 1) Top-Level Page description 250 (Appendix 1); 2) Background.htm (Appendix 2); 3) Icon.htm (Appendix 4); and 4) Name.htm (Appendix4).

[0119] Although the present invention has been described in considerable detail with regard to the preferred versions thereof, other versions are possible.

[0120] The method and system for generating a user interface in a plurality of multiple devices connected to the network system for controlling devices that are connected to a network, according to the present invention can be applied to home networks having multi-media devices connected. The multi-media devices can include PC, VCR, Camcorder, DVD, and HDTV, etc.

Appendix 1- Top-Level Page Example

[0121]

```
<HTML>
<HEAD>
<TITLE>HN Devices Page</TITLE>
</HEAD>

<FRAMESET ROWS="2%, 47%,2%, 22.5%,2%,22.5%, 2%" border=0
color=black>
<NOFRAMES>Sorry does not support frames</NOFRAMES>
<FRAMESET ROWS="100%,0%">
```

```
<FRAME SRC="background.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET COLS="1.2%,23.5%,1.2%,48.2%,1.2%,23.5%,1.2%">
<FRAMESET ROWS="100%,0%">
<FRAME SRC="background.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET ROWS="48%,4%,48%">
<FRAMESET ROWS="73%, 27%">
<FRAME SRC="http://10.1.1.1/icon.htm" SCROLLING="no" NORESIZE>
<FRAME SRC=" http://10.1.1.1/name.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET ROWS="100%,0%">
<FRAME SRC="background.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET ROWS="73%, 27%" >
<FRAME SRC=" http://10.1.1.10/icon.htm" SCROLLING="no" NORESIZE>
<FRAME SRC=" http://10.1.1.10/name.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
</FRAMESET>
<FRAMESET ROWS="100%,0%">
<FRAME SRC="background.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET ROWS="73%, 27%" >
<FRAME SRC=" http://10.1.22.1/icon.htm" SCROLLING="no" NORESIZE>
<FRAME SRC=" http://10.1.22.1/name.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET ROWS="100%,0%">
<FRAME SRC="background.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET ROWS="48%,4%,48%">
<FRAMESET ROWS="73%, 27%" >
<FRAME SRC=" http://10.1.229.1/icon.htm" SCROLLING="no" NORESIZE>
<FRAME SRC=" http://10.1.229.1/name.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET ROWS="100%,0%">
<FRAME SRC="background.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET ROWS="73%, 27%" >
<FRAME SRC=" http://10.30.30.1/icon.htm" SCROLLING="no" NORESIZE>
<FRAME SRC=" http://10.30.30.1/name.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
</FRAMESET>
<FRAMESET ROWS="100%,0%">
<FRAME SRC="background.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
</FRAMESET>
<FRAMESET ROWS="100%,0%">
<FRAME SRC="background.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET
COLS="1.2%,23.5%,1.2%,23.5%,1.2%,23.5%,1.2%,23.5%,1.2%">
<FRAMESET ROWS="100%,0%">
<FRAME SRC="background.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET ROWS="73%, 27%" >
<FRAME SRC=" http://10.41.1.1/icon.htm" SCROLLING="no" NORESIZE>
<FRAME SRC=" http://10.41.1.1/name.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
```

```
<FRAMESET ROWS="100%,0%">
<FRAME SRC="background.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET ROWS="73%, 27%" >
<FRAME SRC=" http://10.41.21.1/icon.htm" SCROLLING="no" NORESIZE>
<FRAME SRC=" http://10.41.21.1/name.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET ROWS="100%,0%">
<FRAME SRC="background.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET ROWS="73%, 27%" >
<FRAME SRC=" http://10.45.1.1/icon.htm" SCROLLING="no" NORESIZE>
<FRAME SRC=" http://10.45.1.1/name.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET ROWS="100%,0%">
<FRAME SRC="background.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET ROWS="73%, 27%" >
<FRAME SRC=" http://10.100.1.1/icon.htm" SCROLLING="no" NORESIZE>
<FRAME SRC=" http://10.100.1.1/name.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET ROWS="100%,0%">
<FRAME SRC="background.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
</FRAMESET>
<FRAMESET ROWS="100%,0%">
<FRAME SRC="background.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET
COLS="1.2%,23.5%,1.2%,23.5%,1.2%,23.5%,1.2%,23.5%,1.2%">
<FRAMESET ROWS="100%,0%">
<FRAME SRC="background.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET ROWS="73%, 27%" >
<FRAME SRC=" http://10.122.22.11/ela.htm" SCROLLING="no" NORESIZE>
<FRAME SRC=" http://10.122.22.1/ela.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET ROWS="100%,0%">
<FRAME SRC="background.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET ROWS="73%, 27%" >
<FRAME SRC=" http://10.122.122.122/icon.htm" SCROLLING="no" NORESIZE>
<FRAME SRC=" http://10.122.122.122/name.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET ROWS="100%,0%">
<FRAME SRC="background.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET ROWS="73%, 27%" >
<FRAME SRC=" http://10.122.122.123/icon.htm" SCROLLING="no" NORESIZE>
<FRAME SRC=" http://10.122.122.123/name.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET ROWS="100%,0%">
<FRAME SRC="background.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
<FRAMESET ROWS="73%, 27%" >
<FRAME SRC=" http://10.122.122.124/icon.htm" SCROLLING="no" . NORESIZE>
<FRAME SRC=" http://10.122.122.1241name.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
```

```
<FRAMESET ROWS="100%,0%">
<FRAME SRC="background.htm" SCROLLING="no" NORESIZE>
</FRAMES ET>
</FRAMESET>
<FRAMESET ROWS="100%,0%">
<FRAME SRC="background.htm" SCROLLING="no" NORESIZE>
</FRAMESET>
</FRAMESET>
<BODY BGCOLOR="#FFFFF0" TEXT="#000070" LINK="#0000ff" ALINK="#FF0000" VLINK="#007986">
</BODY>
</HTML>
```

Appendix 2- Background.htm example

[0122]

```
<HTML>
<HEAD>
<TITLE>Background</TITLE>
</HEAD><BODY BGCOLOR="#007986"></BODY>
</HTML>
```

Appendix 3 - Icon.htm example

[0123]

```
<HTML>
<HEAD>
<TITLE>Device Icon</TITLE>
</HEAD>

<BODY BGCOLOR="#FFFFF0" TEXT="#000070" LINK="#0000ff" ALINK="#FF0000" VLINK="#007986">
<br><br><CENTER>
<IMG SRC="icon.gif" border=0>
</CENTER>
</BODY>

</HTML>
```

Appendix 4 - Name.htm example

[0124]

```
<HTML>
<HEAD>
<TITLE>Device Name</TITLE>
</HEAD>

<BODY BGCOLOR="#FFFFF0" TEXT="#000070" LINK="#0000ff" ALINK="#FF0000" VLINK=#007986">
<CENTER><FONT size=+0>Samsung Device</font></CENTER>
</BODY>
</HTML>
```

Claims

1. A method for providing a user interface (18, 220) for controlling devices (11-14, 102-110, 116, 117, 119, 120) that are currently connected to a network (11 a, 100, 300, 400), the method comprising the steps of:

a) obtaining, by one or more of said devices (12, 102, 103), information from said devices currently connected to the network, said information including device information;
b) generating, in each of said one or more devices, a user interface description (250) wherein the user interface description is generated based at least on the obtained information and includes at least one reference associated with the device information of each of said devices currently connected to the network; and
c) displaying one or more user interfaces (18, 220) each based on one of said one or more user interface descriptions, on one or more devices (12, 102, 103) connected to the network capable of displaying a user interface, for user control of said devices that are currently connected to the network.

2. The method of claim 1, wherein the step of displaying each user interface further includes the steps of:

using each reference in the corresponding user interface description to access the associated information in each device;
generating the user interface including device data corresponding to each device using the accessed information in each device; and
displaying the user interface on said device capable of displaying a user interface.

3. The method of claim 1 or 2, wherein the step of generating a user interface description further comprises the step of associating a hyper-text link with the device information of each of said devices currently connected to the network.

4. The method of one of the claims 1 to 3, wherein said information in each device comprises an HTML page contained in that device.

5. The method of one of the claims 1 to 4, wherein the step of displaying the user interface further comprises the step of displaying the user interface on a browser on said device capable of displaying a user interface.

6. The method of one of the claims 1 to 5, further comprising the steps of:

connecting at least one client device to the network capable of displaying a user interface; and
displaying a user interface on the client device using the references in a user interface description, for controlling devices that are currently connected to the network.

7. The method of claim 1, wherein said at least one device is capable of displaying a user interface, and further comprising the step of displaying a user interface on said at least one device using the references in the user interface description, for controlling devices that are currently connected to the network.

8. The method of one of the claims 1 to 7, wherein the step b) further includes the step of generating each user interface description such that the reference in that user interface description provides access to at least the information in each corresponding device.

9. The method of one of the claims 1 to 8, wherein the step b) further includes the step of generating each user interface description such that the user interface description further includes device data corresponding to each device based on the information obtained from each device.

10. The method of one of the claims 1 to 9, wherein the device information in each device includes device identification information.

11. The method of one of the claims 1 to 10, wherein the device information in each device includes a user control interface description for user interaction with the device.

12. The method of claim 11, wherein the step b) further includes the step of generating each user interface description such that each reference in that user interface description is to at least the user control interface description in each corresponding device.

13. The method of claim 11, wherein the step b) further includes the step of generating each user interface description wherein that user interface description further includes device data corresponding to each device based on the information obtained from each device, the device data providing reference to the user control interface description in each device.

14. A network system (11 a, 100, 300, 400) for performing a service, comprising:

a physical layer (164), wherein the physical layer provides a communication medium than can be used by devices to communicate with each other;
one or more devices (11-14, 102-110, 116, 117, 119, 120) connected to the physical layer, at least one device storing information including device information;
an agent (408, 410) in each of one or more devices (12, 102, 103), adapted for:

a) obtaining, by each of said one or more devices, information from devices currently connected to the network, said information including device information;
b) generating, in each of said one or more devices, a user interface description (250) wherein the user interface description is generated based at least on the obtained information and includes at least one reference associated with the device information of each of said devices currently connected to the network; and
c) displaying one or more user interfaces (18, 220) each based on one of said one or more user interface descriptions, on one or more devices (12, 102, 103) connected to the network capable of displaying a user interface, for user control of said devices that are currently connected to the network.

15. The system of claim 14, wherein information including device information is stored in each of said one or more devices.

16. The system of claim 14 or 15, wherein each agent is further adapted for displaying each user interface by:

using each reference in the corresponding user interface description to access the associated information in each device;
generating the user interface including device data corresponding to each device using the accessed information in each device; and
displaying the user interface on said device capable of displaying a user interface.

17. The system of one of the claims 14 to 16, wherein each agent is further adapted for generating a user interface description by associating a hyper-text link with the device information of each of said devices currently connected to the network.

18. The system of one of the claims 14 to 17, wherein said information in each device comprises an HTML page contained in that device.

19. The system of one of the claims 14 to 18, wherein each agent is further adapted for displaying a user interface by displaying the user interface on a browser on said device capable of displaying a user interface.

20. The system of one of the claims 14 to 19, further comprising at least one client device connected to the network capable of displaying a user interface; and
wherein one or more agents are further adapted for displaying a user interface on the client device using the references in a corresponding user interface description, for controlling devices that are currently connected to the network.

21. The system of claim 14, wherein said at least one device is capable of displaying a user interface, and one or more agents are further adapted for displaying a user interface on said at least one device using the references in a corresponding user interface description, for controlling devices that are currently connected to the network.

22. The system of one of the claims 14 to 21, wherein each agent is further adapted for generating each user interface description such that the reference in that user interface description provides access to at least the information in each corresponding device.

23. The system of one of the claims 14 to 22, wherein each agent is further adapted for generating each user interface description such that the user interface description further includes device data corresponding to each device based on the information obtained from each device.

24. The system of one of the claims 14 to 23, wherein the device information in each device includes device identification information.

25. The system of one of the claims 14 to 24, wherein the device information in each device includes a user control interface description for user interaction with the device.

26. The system of claim 25, wherein each agent is further adapted for generating each user interface description such that each reference in that user interface description is to at least the user control interface description in each corresponding device.

27. The system of claim 25, wherein each agent is further adapted for generating each user interface description wherein that user interface description further includes device data corresponding to each device based on the information obtained from each device, the device data providing reference to the user control interface description in each device.

28. The system of one of the claims 14 to 27, wherein, if multiple devices are connected to the physical layer, one or more of said multiple devices store information including device information, and a plurality of said multiple devices each include an agent.

Patentansprüche

1. Verfahren zur Bereitstellung einer Benutzerschnittstelle (18, 220) zur Steuerung von Geräten (11-14, 102-110, 116, 117, 119, 120), die gerade mit einem Netzwerk (11a, 100, 300, 400) verbunden sind, umfassend folgende Schritte:

   a) Erhalten von Informationen durch eines oder mehrere der Geräte (12, 102, 103) von den Geräten, die gerade mit dem Netzwerk verbunden sind, wobei die Informationen Geräteinformationen enthalten;
   b) Generieren einer Benutzerschnittstellenbeschreibung in jedem der ein oder mehreren Geräte, wobei die Benutzerschnittstellenbeschreibung auf Basis zumindest der erhaltenen Informationen generiert wird und zumindest eine mit den Geräteinformationen jedes der gerade mit dem Netzwerk verbundenen Geräte assoziierte Referenz enthält; und
   c) Anzeigen einer oder mehrerer Benutzerschnittstellen (18, 220), die jeweils auf einer der einen oder mehreren Benutzerschnittstellenbeschreibungen basieren, auf einem oder mehreren mit dem Netzwerk verbundenen Geräten (12, 102, 103), die in der Lage sind, eine Benutzerschnittstelle anzuzeigen, zur Benutzersteuerung der Geräte, die gerade mit dem Netzwerk verbunden sind.

2. Verfahren nach Anspruch 1, wobei der Schritt des Anzeigens jeder Benutzerschnittstelle weiterhin folgende Schritte umfasst:

   Verwenden jeder Referenz in der entsprechenden Benutzerschnittstellenbeschreibung, um auf die assoziierten Informationen in jedem Gerät zuzugreifen;
   Generieren der Benutzerschnittstelle einschließlich Gerätedaten entsprechend jedem Gerät unter Benutzung der Informationen in jedem Gerät, auf die zugegriffen wurde;
   Anzeigen der Benutzerschnittstelle auf dem Gerät, das in der Lage ist, eine Benutzerschnittstelle anzuzeigen.

3. Verfahren nach Anspruch 1 oder 2, wobei der Schritt des Generierens einer Benutzerschnittstellenbeschreibung weiterhin einen Schritt des Assoziierens eines Hypertext-Links mit den Geräteinformationen jedes der Geräte, die gerade mit dem Netzwerk verbunden sind, umfasst.

4. Verfahren nach einem der Ansprüche 1 bis 3, wobei die Informationen in jedem Gerät eine in dem Gerät enthaltene HTML-Seite umfassen.

5. Verfahren nach einem der Ansprüche 1 bis 4, wobei der Schritt des Anzeigens der Benutzerschnittstelle weiterhin einen Schritt des Anzeigens der Benutzerschnittstelle auf einem Browser auf dem Gerät, das in der Lage ist, eine Benutzerschnittstelle anzuzeigen, umfasst.

6. Verfahren nach einem der Ansprüche 1 bis 5, weiterhin umfassend folgende Schritte:

   Verbinden mindestens eines Client-Geräts mit dem Netzwerk, das in der Lage ist, eine Benutzerschnittstelle anzuzeigen; und
   Anzeigen einer Benutzerschnittstelle auf dem Client-Gerät unter Benutzung der Referenzen in einer Benutzerschnittstellenbeschreibung zur Steuerung von Geräten, die gerade mit dem Netzwerk verbunden sind.

7. Verfahren nach Anspruch 1, wobei das mindestens eine Gerät in der Lage ist, eine Benutzerschnittstelle anzuzeigen, und weiterhin umfassend einen Schritt des Anzeigens einer Benutzerschnittstelle auf dem mindestens einen Gerät unter Benutzung der Referenzen in der Benutzerschnittstellenbeschreibung zur Steuerung von Geräten, die gerade mit dem Netzwerk verbunden sind.

8. Verfahren nach einem der Ansprüche 1 bis 7, wobei Schritt b) weiterhin einen Schritt des Generierens jeder Benutzerschnittstellenbeschreibung so umfasst, dass die Referenz in dieser Benutzerschnittstellenbeschreibung Zugriff auf zumindest die Informationen in jedem entsprechenden Gerät bereitstellt.

9. Verfahren nach einem der Ansprüche 1 bis 8, wobei Schritt b) weiterhin einen Schritt des Generierens jeder Benutzerschnittstellenbeschreibung so umfasst, dass auf Basis der von jedem Gerät erhaltenen Informationen die Benutzerschnittstellenbeschreibung weiterhin Gerätedaten entsprechend jedem Gerät enthält.

10. Verfahren nach einem der Ansprüche 1 bis 9, wobei die Geräteinformationen in jedem Gerät Geräteidentifikationsinformationen enthalten.

11. Verfahren nach einem der Ansprüche 1 bis 10, wobei die Geräteinformationen in jedem Gerät eine Benutzersteuerschnittstellenbeschreibung für eine Benutzerinteraktion mit dem Gerät enthalten.

12. Verfahren nach Anspruch 11, wobei Schritt b) weiterhin einen Schritt des Generierens jeder Benutzerschnittstellenbeschreibung so umfasst, dass jede Referenz in dieser Benutzerschnittstellenbeschreibung sich auf zumindest die Benutzersteuerschnittstellenbeschreibung in jedem entsprechenden Gerät bezieht.

13. Verfahren nach Anspruch 11, wobei Schritt b) weiterhin einen Schritt des Generierens jeder Benutzerschnittstellenbeschreibung umfasst, wobei diese Benutzerschnittstellenbeschreibung weiterhin Gerätedaten enthält, die jedem Gerät auf Basis von von jedem Gerät erhaltenen Informationen entspricht, wobei Gerätedaten Referenz auf die Benutzersteuerschnittstellenbeschreibung in jedem Gerät bereitstellen.

14. Netzwerksystem (11a, 100, 300, 400) zum Durchführen eines Dienstes und umfassend:

eine physikalische Schicht (164), wobei die physikalische Schicht ein Kommunikationsmedium bereitstellt, das durch Geräte zum Kommunizieren miteinander verwendet werden kann;
ein oder mehrere Geräte (11-14, 102-110, 116, 117, 119, 120), die mit der physikalischen Schicht verbunden sind, wobei mindestens ein Gerät Informationen speichert, die Geräteinformationen enthalten;
einen Agent (408, 410) (Vermittler, Mittel, "agent") in jedem der ein oder mehreren Geräte (12, 102, 103), der zu Folgendem angepasst ist:

a) Erhalten von Informationen durch jedes der ein oder mehreren Geräte von Geräten, die gerade mit dem Netzwerk verbunden sind, wobei die Informationen Geräteinformationen enthalten;
b) Generieren einer Benutzerschnittstellenbeschreibung in jedem der ein oder mehreren Geräte, wobei die Benutzerschnittstellenbeschreibung auf Basis zumindest der erhaltenen Informationen generiert wird und zumindest eine mit der Geräteinformation jedes der gerade mit dem Netzwerk verbundenen Geräte assoziierte Referenz enthält; und
c) Anzeigen einer oder mehrerer Benutzerschnittstellen (18, 220), die jeweils auf einer der ein oder mehreren Benutzerschnittstellenbeschreibungen basieren, auf einem oder mehreren mit dem Netzwerk verbundenen Geräten (12, 102, 103), die in der Lage sind, eine Benutzerschnittstelle anzuzeigen, zur Benutzersteuerung der Geräte, die gerade mit dem Netzwerk verbunden sind.

15. System nach Anspruch 14, wobei die Informationen, die Geräteinformationen enthalten, in jedem der ein oder mehreren Geräte gespeichert ist.

16. System nach Anspruch 14 oder 15, wobei jeder Agent weiterhin dazu angepasst ist, jede Benutzerschnittstelle durch folgende Schritte anzuzeigen:

Benutzen jeder Referenz in der entsprechenden Benutzerschnittstellenbeschreibung, um auf die assoziierten Informationen in jedem Gerät zuzugreifen;
Generieren der Benutzerschnittstelle, die jedem Gerät entsprechende Grätedaten enthält, unter Benutzung der Informationen in jedem Gerät, auf die zugegriffen wurde; und

Anzeigen der Benutzerschnittstelle auf dem Gerät, das in der Lage ist, eine Benutzerschnittstelle anzuzeigen.

17. System nach einem der Ansprüche 14 bis 16, wobei jeder Agent ferner dazu angepasst ist, eine Benutzerschnittstellenbeschreibung durch Assoziieren eines Hypertext-Links mit den Geräteinformationen jedes der Geräte, die gerade mit dem Netzwerk verbunden sind, zu generieren.

18. System nach einem der Ansprüche 14 bis 17, wobei die Informationen in jedem Gerät eine in diesem Gerät enthaltene HTML-Seite umfassen.

19. System nach einem der Ansprüche 14 bis 18, wobei jeder Agent weiterhin dazu angepasst ist, eine Benutzerschnittstelle durch Anzeigen der Benutzerschnittstelle auf einem Browser auf dem Gerät, das in der Lage ist, eine Benutzerschnittstelle anzuzeigen, anzuzeigen.

20. System nach einem der Ansprüche 14 bis 19, weiterhin umfassend zumindest ein Client-Gerät, das mit dem Netzwerk verbunden ist und in der Lage ist, eine Benutzerschnittstelle anzuzeigen; und
wobei ein oder mehrere Agents weiterhin dazu angepasst sind, eine Benutzerschnittstelle auf dem Client-Gerät unter Benutzung der Referenzen in einer entsprechenden Benutzerschnittstellenbeschreibung anzuzeigen, um Geräte zu steuern, die gerade mit dem Netzwerk verbunden sind.

21. System nach Anspruch 14, wobei das mindestens eine Gerät in der Lage ist, eine Benutzerschnittstelle anzuzeigen, und ein oder mehrere Agents weiterhin dazu angepasst sind, eine Benutzerschnittstelle auf dem zumindest einen Gerät unter Benutzung der Referenzen in einer entsprechenden Benutzerschnittstellenbeschreibung anzuzeigen, um Geräte zu steuern, die gerade mit dem Netzwerk verbunden sind.

22. System nach einem der Ansprüche 14 bis 21, wobei jeder Agent weiterhin dazu angepasst ist, jede Benutzerschnittstellenbeschreibung so zu generieren, dass die Referenz in dieser Benutzerschnittstellenbeschreibung Zugriff auf zumindest die Informationen in jedem entsprechenden Gerät bereitstellt.

23. System nach einem der Ansprüche 14 bis 22, wobei jeder Agent weiterhin dazu angepasst ist, jede Benutzerschnittstellenbeschreibung so zu generieren, dass die Benutzerschnittstellenbeschreibung weiterhin Gerätedaten auf Basis der von jedem Gerät erhaltenen Informationen enthält, die jedem Gerät entsprechen.

24. System nach einem der Ansprüche 14 bis 23, wobei die Geräteinformationen in jedem Gerät Geräteidentifikationsinformationen enthalten.

25. System nach einem der Ansprüche 14 bis 24, wobei die Geräteinformationen in jedem Gerät eine Benutzersteuerschnittstellenbeschreibung für eine Benutzerinteraktion mit dem Gerät enthalten.

26. System nach Anspruch 25, wobei jeder Agent weiterhin dazu angepasst ist, jede Benutzerschnittstellenbeschreibung so zu generieren, dass jede Referenz in dieser Benutzerschnittstellenbeschreibung sich zumindest auf die Benutzersteuerschnittstellenbeschreibung in jedem entsprechenden Gerät bezieht.

27. System nach Anspruch 25, wobei jeder Agent weiterhin dazu angepasst ist, jede Benutzerschnittstellenbeschreibung zu generieren, wobei diese Benutzerschnittstellenbeschreibung weiterhin Gerätedaten enthält, die jedem Gerät auf Basis von von jedem Gerät erhaltenen Informationen entsprechen, wobei die Gerätedatenreferenz auf die Benutzersteuerschnittstellenbeschreibung in jedem Gerät bereitstellen.

28. System nach einem der Ansprüche 14 bis 27, wobei, wenn mehrere Geräte mit der physikalischen Schicht verbunden sind, eines oder mehrere der mehreren Geräte Informationen speichern, die Geräteinformationen enthalten, und wobei eine Vielzahl der mehreren Geräte jeweils einen Agent enthalten.

Revendications

1. Procédé pour fournir une interface utilisateur (18, 220) pour commander des dispositifs (11-14, 102-110, 116, 117, 119, 120) qui sont actuellement connectés à un réseau (11a, 100, 300, 400), le procédé comportant les étapes consistant à :

a) obtenir, par l'intermédiaire d'un ou plusieurs dits dispositifs (12, 102, 103), des informations en provenance dits dispositifs actuellement connectés au réseau, lesdites informations incluant des informations de dispositif,
b) générer, dans chacun dudit ou desdits dispositifs, une description d'interface utilisateur (250), la description d'interface utilisateur étant générée sur la base d'au moins les informations obtenues et incluant au moins une référence associée aux informations de dispositif de chacun desdits dispositifs actuellement connectés au réseau, et
c) afficher une ou plusieurs interfaces utilisateur (18, 220) chacune étant basée sur une parmi ladite ou lesdites descriptions d'interface utilisateur, sur un ou plusieurs dispositifs (12, 102, 103) connectés au réseau capables d'afficher une interface utilisateur, pour une commande d'utilisateur desdits dispositifs qui sont actuellement connectés au réseau.

2. Procédé selon la revendication 1, dans lequel l'étape consistant à afficher chaque interface utilisateur inclut en outre les étapes consistant à :

utiliser chaque référence dans la description d'interface utilisateur correspondante pour accéder aux informations associées dans chaque dispositif,
générer l'interface utilisateur incluant des données de dispositif correspondant à chaque dispositif utilisant les informations ayant fait l'objet d'un accès dans chaque dispositif, et
afficher l'interface utilisateur sur ledit dispositif capable d'afficher une interface utilisateur.

3. Procédé selon la revendication 1 ou 2, dans lequel l'étape consistant à générer une description d'interface utilisateur comporte en outre l'étape consistant à associer un lien hypertexte aux informations de dispositif de chacun desdits dispositifs actuellement connecté au réseau.

4. Procédé selon l'une des revendications 1 à 3, dans lequel lesdites informations dans chaque dispositif comportent une page HTML contenue dans ce dispositif.

5. Procédé selon l'une des revendications 1 à 4, dans lequel l'étape consistant à afficher l'interface utilisateur comporte en outre l'étape consistant à afficher l'interface utilisateur sur un navigateur sur ledit dispositif capable d'afficher une interface utilisateur.

6. Procédé selon l'une des revendications 1 à 5, comportant en outre les étapes consistant à :

connecter au moins un dispositif client au réseau capable d'afficher une interface utilisateur, et
afficher une interface utilisateur sur le dispositif client en utilisant les références d'une description d'interface utilisateur, pour commander des dispositifs qui sont actuellement connectés au réseau.

7. Procédé selon la revendication 1, dans lequel ledit au moins un dispositif est capable d'afficher une interface utilisateur, et comportant en outre l'étape consistant à afficher une interface utilisateur sur ledit au moins un dispositif en utilisant les références de la description d'interface utilisateur, pour commander des dispositifs qui sont actuellement connectés au réseau.

8. Procédé selon l'une des revendications 1 à 7, dans lequel l'étape b) inclut en outre l'étape consistant à générer chaque description d'interface utilisateur de sorte que la référence dans cette description d'interface utilisateur fournit l'accès à au moins les informations dans chaque dispositif correspondant.

9. Procédé selon l'une des revendications 1 à 8, dans lequel l'étape b) inclut en outre l'étape consistant à générer chaque description d'interface utilisateur de sorte que la description d'interface utilisateur inclut en outre des données de dispositif correspondant à chaque dispositif sur la base des informations obtenues en provenance de chaque dispositif.

10. Procédé selon l'une des revendications 1 à 9, dans lequel les informations de dispositif dans chaque dispositif incluent des informations d'identification de dispositif.

11. Procédé selon l'une des revendications 1 à 10, dans lequel les informations de dispositif de chaque dispositif incluent une description d'interface de commande utilisateur pour une interaction d'utilisateur avec le dispositif.

12. Procédé selon la revendication 11, dans lequel l'étape b) inclut en outre l'étape consistant à générer chaque description d'interface utilisateur de sorte que chaque référence dans cette description d'interface utilisateur va au moins jusqu'à la description d'interface de commande utilisateur dans chaque dispositif correspondant.

13. Procédé selon la revendication 11, dans lequel l'étape b) inclut en outre l'étape consistant à générer chaque description d'interface utilisateur, cette description d'interface utilisateur incluant en outre des données de dispositif correspondant à chaque dispositif sur la base des informations obtenues en provenance de chaque dispositif, les données de dispositif fournissant une référence à la description d'interface de commande utilisateur dans chaque dispositif.

14. Système de réseau (11a, 100, 300, 400) pour effectuer un service, comportant :

une couche physique (164), la couche physique fournissant un support de communication qui peut être utilisé par des dispositifs pour communiquer entre eux,
un ou plusieurs dispositifs (11-14, 102-110, 116, 117, 119, 120) connectés à la couche physique, au moins un dispositif mémorisant des informations incluant des informations de dispositif,
un agent (408, 410) dans chacun du ou des dispositifs (12, 102, 103), adapté pour :

a) obtenir, par l'intermédiaire de chacun dudit ou desdits dispositifs, des informations en provenance des dispositifs actuellement connectés au réseau, lesdites informations incluant des informations de dispositif,
b) générer, dans chacun dudit ou desdits dispositifs, une description d'interface utilisateur (250), la description d'interface utilisateur étant générée sur la base d'au moins les informations obtenues et incluant au moins une référence associée aux informations de dispositif de chacun desdits dispositifs actuellement connectés au réseau, et
c) afficher une ou plusieurs interfaces utilisateur (18, 220) chacune étant basée sur l'une de ladite ou desdites descriptions d'interface utilisateur, sur un ou plusieurs dispositifs (12, 102, 103) connectés au réseau capables d'afficher une interface utilisateur, pour une commande d'utilisateur desdits dispositifs qui sont actuellement connectés au réseau.

15. Système selon la revendication 14, dans lequel des informations incluant des informations de dispositif sont mémorisées dans chacun dudit ou desdits dispositifs .

16. Système selon la revendication 14 ou 15, dans lequel chaque agent est en outre adapté pour afficher chaque interface utilisateur en :

utilisant chaque référence dans la description d'interface utilisateur correspondante pour accéder aux informations associées dans chaque dispositif,
générant l'interface utilisateur incluant des données de dispositif correspondant à chaque dispositif utilisant les informations ayant fait l'objet d'un accès dans chaque dispositif, et
affichant l'interface utilisateur sur ledit dispositif capable d'afficher une interface utilisateur.

17. Système selon l'une quelconque des revendications 14 à 16, dans lequel chaque agent est en outre adapté pour générer une description d'interface utilisateur en associant un lien hypertexte au réseau.

18. Système selon l'une quelconque des revendications 14 à 17, dans lequel lesdites informations dans chaque dispositif comportent une page HTML contenue dans ce dispositif.

19. Système selon l'une quelconque des revendications 14 à 18, dans lequel chaque agent est en outre adapté pour afficher une interface utilisateur en affichant l'interface utilisateur sur un navigateur sur ledit dispositif capable d'afficher une interface utilisateur.

20. Système selon l'une des revendications 14 à 19, comportant en outre au moins un dispositif client connecté au réseau capable d'afficher une interface utilisateur, et
dans lequel un ou plusieurs agents sont en outre adaptés pour afficher une interface utilisateur sur le dispositif client en utilisant les références dans une description d'interface utilisateur correspondante, pour commander des dispositifs qui sont actuellement connectés au réseau.

21. Système selon la revendication 14, dans lequel ledit au moins un dispositif est capable d'afficher une interface

utilisateur, et un ou plusieurs agents sont en outre adaptés pour afficher une interface utilisateur sur ledit au moins un dispositif en utilisant les références dans une description d'interface utilisateur correspondante, pour commander des dispositifs qui sont actuellement connectés au réseau.

22. Système selon l'une des revendications 14 à 21, dans lequel chaque agent est en outre adapté pour générer chaque description d'interface utilisateur de sorte que la référence dans cette description d'interface utilisateur fournit un accès au moins aux informations dans chaque dispositif correspondant.

23. Système selon l'une des revendications 14 à 22, dans lequel chaque agent est en outre adapté pour générer chaque description d'interface utilisateur de sorte que la description d'interface utilisateur inclut en outre des données de dispositif correspondant à chaque dispositif sur la base des informations obtenues en provenance de chaque dispositif.

24. Système selon l'une des revendications 14 à 23, dans lequel les informations de dispositif dans chaque dispositif incluent des informations d'identification de dispositif.

25. Système selon l'une des revendications 14 à 24, dans lequel les informations de dispositif de chaque dispositif incluent une description d'interface de commande utilisateur pour une interaction d'utilisateur avec le dispositif.

26. Système selon la revendication 25, dans lequel chaque agent est en outre adapté pour générer chaque description d'interface utilisateur de sorte que chaque référence dans cette description d'interface utilisateur va au moins jusqu'à la description d'interface de commande utilisateur dans chaque dispositif correspondant.

27. Système selon la revendication 25n dans lequel chaque agent est en outre adapté pour générer chaque description d'interface utilisateur, dans lequel cette description d'interface utilisateur inclut en outre des données de dispositif correspondant à chaque dispositif sur la base des informations obtenues en provenance de chaque dispositif, les données de dispositif fournissant une référence à la description d'interface de commande d'utilisateur dans chaque dispositif.

28. Système selon l'une des revendications 14 à 27, dans lequel, si de multiples dispositifs sont connectés à la couche physique, un ou plusieurs desdits multiples dispositifs mémorisent des informations incluant des informations de dispositif, et une pluralité desdits multiples dispositifs inclus tous à un agent.

FIG. 1

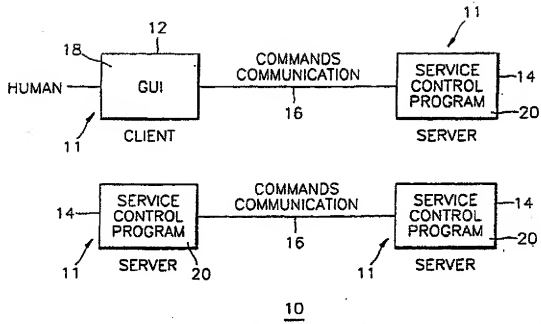


FIG. 2

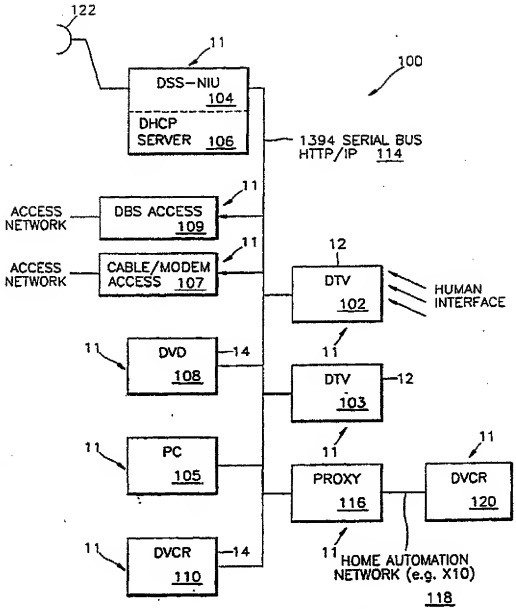


FIG. 3

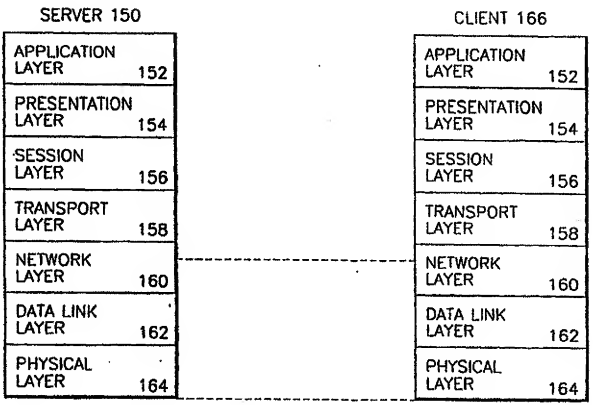

SERVER 150
APPLICATION LAYER 152
PRESENTATION LAYER 154
SESSION LAYER 156
TRANSPORT LAYER 158
NETWORK LAYER 160
DATA LINK LAYER 162
PHYSICAL LAYER 164
CLIENT 166
APPLICATION LAYER 152
PRESENTATION LAYER 154
SESSION LAYER 156
TRANSPORT LAYER 158
NETWORK LAYER 160
DATA LINK LAYER 162
PHYSICAL LAYER 164

FIG. 4A

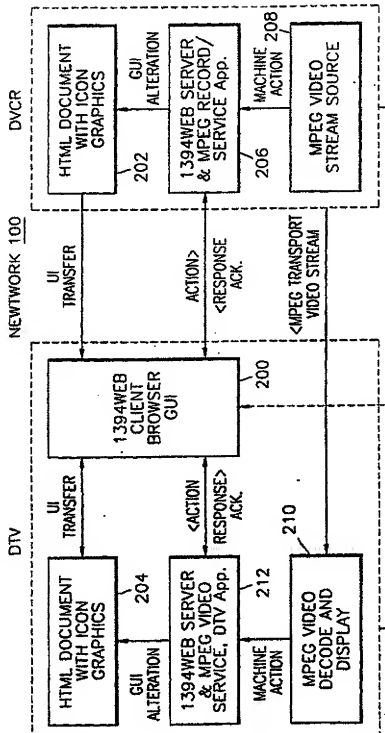


FIG. 4B

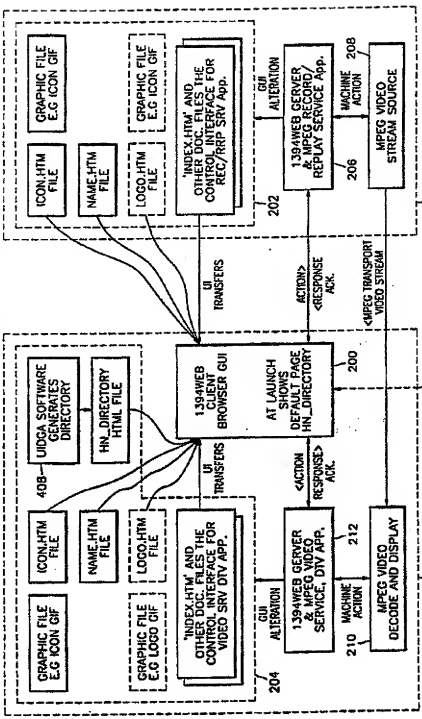


FIG. 5

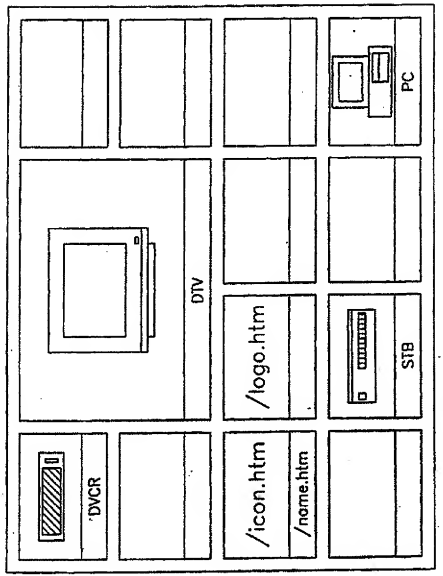


FIG. 6

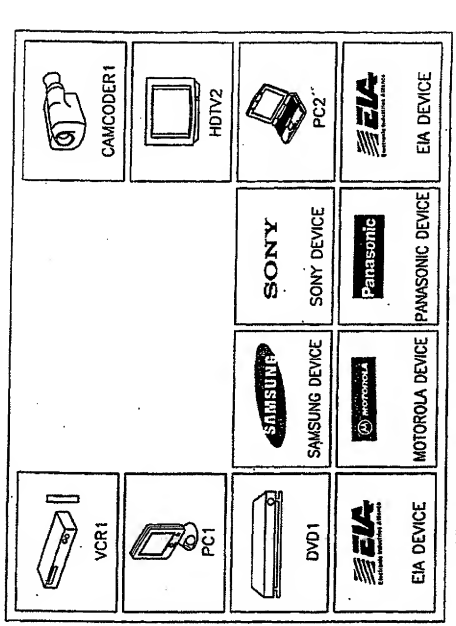


FIG. 7

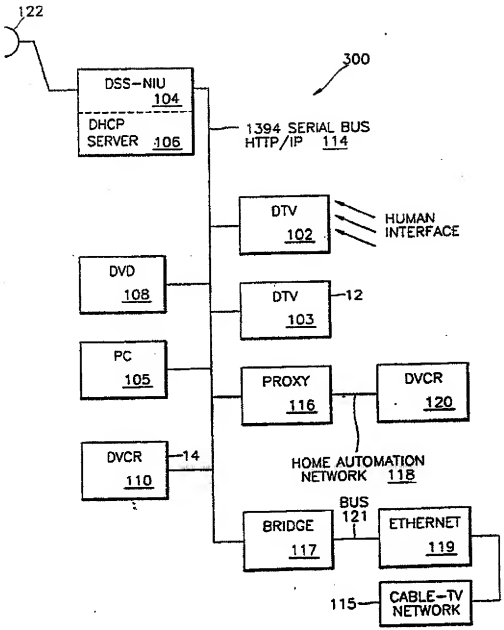


FIG. 8

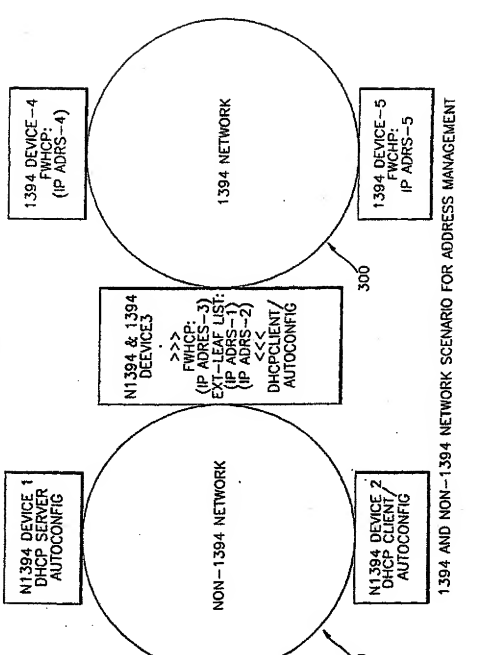


FIG. 9A

FIG. 9B

FIG. 9C

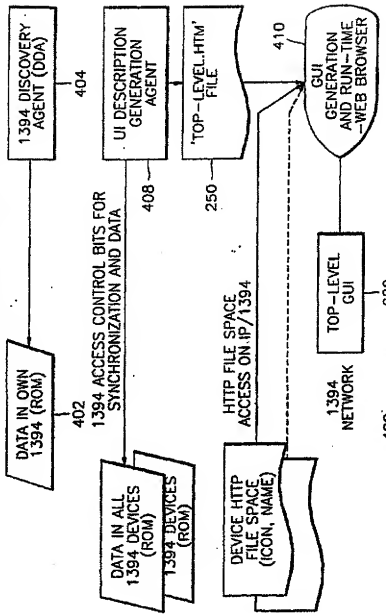


FIG. 10

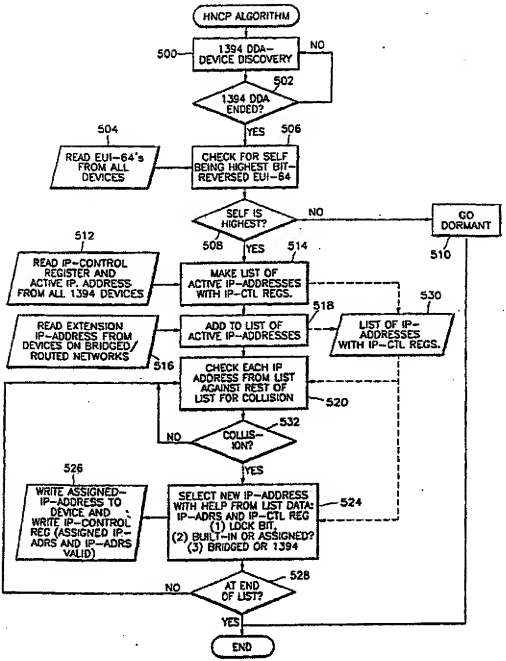
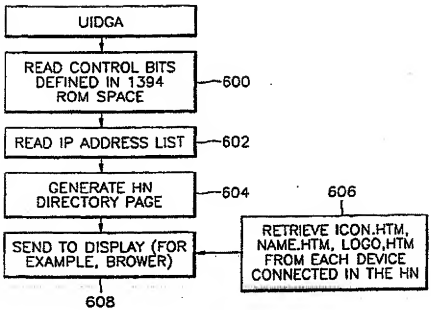


FIG. 11



REFERENCES CITED IN THE DESCRIPTION

This list of references cited by the applicant is for the reader's convenience only. It does not form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.

Patent documents cited in the description

- WO 9859282 A [0010]
- WO 9957639 A [0011]